



UNIVERSIDADE FEDERAL DE SERGIPE  
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

## **Modelos de Alocação de Banda com Justiça em Controladores de Redes Definidas por Software**

Dissertação de Mestrado

José Andeson Moraes de Oliveira



São Cristóvão – Sergipe

2021

UNIVERSIDADE FEDERAL DE SERGIPE  
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

José Andeson Moraes de Oliveira

## **Modelos de Alocação de Banda com Justiça em Controladores de Redes Definidas por Software**

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Sergipe como requisito parcial para a obtenção do título de mestre em Ciência da Computação.

Orientador(a): Prof. Dr. Ricardo José Paiva de Britto Salgueiro

São Cristóvão – Sergipe

2021




**UNIVERSIDADE FEDERAL DE SERGIPE**  
**PRÓ-REITORIA DE PÓS-GRADUAÇÃO E PESQUISA**  
**COORDENAÇÃO DE PÓS-GRADUAÇÃO**  
**PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO**

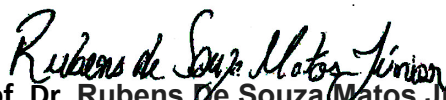
---

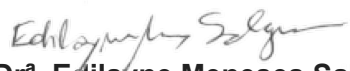
**Ata da Sessão Solene de Defesa da Dissertação do**  
**Curso de Mestrado em Ciência da Computação-UFS.**  
**Candidato: JOSÉ ANDESON MORAIS DE OLIVEIRA**

Em 27 dias do mês de novembro do ano de dois mil e vinte, com início às 15h00min, realizou-se na Sala virtual <https://meet.google.com/jtf-armp-sqm>. A Sessão Pública de Defesa de Dissertação de Mestrado do candidato **José Andeson Moraes De Oliveira**, que desenvolveu o trabalho intitulado: **“Modelos de alocação de banda com justiça em controladores de rede definidas por software”**, sob a orientação do Prof. Dr. **Ricardo José Paiva De Britto Salgueiro**. A Sessão foi presidida pelo Prof. Dr. **Ricardo José Paiva De Britto Salgueiro** (PROCC/UFS), que após a apresentação da dissertação passou a palavra aos outros membros da Banca Examinadora, Prof. Dr. **Rubens De Souza Matos Junior** (PROCC/UFS), Prof<sup>a</sup>. Dr<sup>a</sup>. **Edilayne Meneses Salgueiro** (Dcomp/UFS)e, em seguida, ao Prof<sup>a</sup>. Dr<sup>a</sup>. **Fernanda Maria Ribeiro De Alencar** (UFPE). Após as discussões, a Banca Examinadora reuniu-se e considerou o mestrando (a ) aprovado “(aprovado/reprovado)”. Atendidas as exigências da Instrução Normativa 01/2017/PROCC, do Regimento Interno do PROCC (Resolução 67/2014/CONEPE), Resolução nº 25/2014/CONEPE e da Portaria nº 413 de 27 de maio de 2020 ( Banca por videoconferência) que regulamentam a Apresentação e Defesa de Dissertação, e nada mais havendo a tratar, a Banca Examinadora elaborou esta Ata que será assinada pelos seus membros e pelo mestrando.

Cidade Universitária “Prof. José Aloísio de Campos”, 27 de novembro de 2020.

  
**Prof. Dr. Ricardo José Paiva De Britto Salgueiro**  
**(PROCC/UFS)**  
**Presidente**

  
**Prof. Dr. Rubens De Souza Matos Junior**  
**(PROCC/UFS)**  
**Examinador Interno**

  
**Prof<sup>a</sup>. Dr<sup>a</sup>. Edilayne Meneses Salgueiro**  
**(Dcomp/UFS)**  
**Examinador Interno à instituição**

  
**Prof<sup>a</sup>. Dr<sup>a</sup>. Fernanda Maria Ribeiro De Alencar**  
**(UFPE/UFS)**  
**Examinador Externo**

  
**JOSÉ ANDESON MORAIS DE OLIVEIRA**  
**discente**

*Dedico este trabalho à minha família, em especial a meus pais. Vocês são e sempre serão minha maior fonte de inspiração e de orgulho.*

# Agradecimentos

Agradeço primeiramente a Deus, por me conceder a oportunidade de chegar até aqui e a força para concluir mais essa etapa em minha vida. Sem Sua permissão e bondade nada seria possível.

Agradeço aos meus pais, por tudo que fizeram por mim durante toda minha vida. Pessoas simples e humildes que nunca tiveram a oportunidade de estudar mas sempre fizeram o possível para que eu e meus irmãos tivéssemos as oportunidades que lhes faltaram e sempre nos incentivou a nos dedicarmos aos estudos.

Agradeço aos meus irmãos, Júnior, Amanda e Edênia por todo apoio e incentivo que sempre me deram ao longo de toda minha vida, em todos os momentos.

Agradeço ao amigo Glauco, pela ajuda que me deu em vários momentos durante a minha jornada acadêmica e também pessoal, por ter sido o grande incentivador para que eu me interessasse a ingressar no mestrado acadêmico.

Agradeço à minha namorada Eliana, pelo companheirismo, compreensão e ajuda que tem me dado ao longo desse período.

Agradeço de forma especial ao meu orientador, Prof. Dr. Ricardo José Paiva de Britto Salgueiro, à Prof<sup>a</sup>. Dra. Edilayne Meneses Salgueiro por todos os ensinamentos e suporte durante todo o decorrer do programa de mestrado.

Agradeço a todos os meus colegas de mestrado que de alguma forma contribuíram com a realização deste trabalho e, conseqüentemente, para minha formação.

Agradeço também ao meu amigo Beto (*in memoriam*), que se foi muito cedo e de forma inesperada, mas participou do início do estudo que deu origem a este trabalho.

A todos vocês, muito obrigado!

*"Eu acredito que às vezes são as pessoas que ninguém espera nada que fazem as coisas que  
ninguém consegue imaginar."*

*Alan Turing*

# Resumo

As mídias sociais, dispositivos móveis e a computação em nuvem estão cada vez mais forçando as redes de computadores convencionais para seus limites. O aumento de poder de processamento e capacidade de armazenamento têm trazido benefícios às infraestruturas com grandes inovações em virtualização e automação, mas estes benefícios muitas vezes são limitados pelas redes. Diante das limitações enfrentadas pelos administradores de redes, o conceito de Redes Definidas por Software (SDN - *Software-Defined Networking*) surgiu como uma proposta de abordagem mais dinâmica, gerenciável e adaptável, onde os planos de dados e de controle são desacoplados. Com este desacoplamento, onde o plano de controle é centralizado e o plano de dados fica em cada dispositivo, a rede torna-se mais flexível, eficiente e ideal para alta largura de banda e para a natureza dinâmica dos aplicativos atuais. Outro aspecto importante é a alocação de recursos realizada por alguns elementos da rede para atender à demanda de aplicações. Mecanismos de alocação devem ser capazes de atribuir de maneira eficiente os recursos disponíveis em uma rede entre as diferentes aplicações que competem por eles. Neste sentido, devido à padronização e adaptabilidade propostas pelas SDNs, e das limitações na arquitetura das redes convencionais, esta dissertação apresenta uma solução que permite utilizar modelos de alocação de recursos com justiça em um controlador de rede baseado na arquitetura SDN a fim de otimizar mecanismos para compartilhamento de banda. Para validação das implementações realizadas, foram executados experimentos em ambiente virtual e em um laboratório de experimentos. Os resultados alcançados foram considerados satisfatórios, comprovando que o compartilhamento de banda em redes SDN promove uma maior eficiência na utilização dos recursos disponíveis.

**Palavras-chave:** SDN, redes, QoS, compartilhamento, justiça, OpenFlow, controlador, largura de banda, Floodlight.

# Abstract

Social media, mobile devices and cloud computing are increasingly pushing conventional computer networks to their limits. Increased processing power and storage capacity have brought benefits to infrastructures with major innovations in virtualization and automation, but these benefits are often limited by networks. Given the limitations faced by network administrators, the concept of Software-Defined Networking (SDN) has emerged as a proposal for a more dynamic, manageable and adaptable approach where data and control plans are decoupled. With this uncoupling, where the control plan is centralized and the data plan is on each device, the network becomes more flexible, efficient, and ideal for high bandwidth and the dynamic nature of today's applications. Another important aspect is the allocation of resources performed by some elements of the network to meet the demand for applications. Allocation mechanisms must be able to efficiently allocate the resources available on a network among the different applications that compete for them. In this sense, due to the standardization and adaptability proposed by the SDNs, and the limitations in the architecture of conventional networks, this dissertation proposal proposes a solution that allows the use of resource allocation models with justice in a network controller based on the SDN architecture in order to optimize mechanisms for band sharing. To validate this technique, experiments will be carried out in a virtual environment and in an experiment lab

**Keywords:** SDN, networking, QoS, sharing, fairness, OpenFlow, controller, bandwidth, Flood-light.



# Lista de ilustrações

Figura 1 – Etapas da Metodologia da Pesquisa . . . . .	21
Figura 2 – Detalhes Arquitetura SDN . . . . .	24
Figura 3 – Controlador SDN . . . . .	24
Figura 4 – Arquitetura do Controlador SDN. . . . .	25
Figura 5 – Maximum Allocation Model - (MAM) . . . . .	28
Figura 6 – Maximum Allocation with Reservation - (MAR) . . . . .	29
Figura 7 – Russian Doll Model - (RDM) . . . . .	29
Figura 8 – Regra de alocação MARS . . . . .	31
Figura 9 – Regra de alocação SHARM . . . . .	31
Figura 10 – Fases de uma revisão sistemática . . . . .	34
Figura 11 – Trabalhos encontrados por base de busca. . . . .	37
Figura 12 – Quantidade de estudos selecionados por base de busca. . . . .	39
Figura 13 – Quantidade de trabalhos selecionados por ano de publicação. . . . .	40
Figura 14 – Quantidade de citações por publicação. . . . .	41
Figura 15 – Resposta <b>QP1</b> . . . . .	43
Figura 16 – Resposta <b>QP2</b> . . . . .	43
Figura 17 – Questões de pesquisa e a porcentagem de trabalhos que as respondem . . . . .	44
Figura 18 – Representação canônica MAMDFS (Autor) . . . . .	52
Figura 19 – Representação canônica MAMDFS BCs realocadas (Autor) . . . . .	52
Figura 20 – Regra de Alocação MAMDFS (Autor) . . . . .	53
Figura 21 – Visão macro da arquitetura (Autor). . . . .	55
Figura 22 – Componentes de arquitetura <i>BW-Manager</i> (Autor). . . . .	56
Figura 23 – Diagrama de atividades do módulo <i>BW-Manager</i> . . . . .	58
Figura 24 – Topologia experimentos . . . . .	64
Figura 25 – Tamanho da filas antes da perda de pacotes . . . . .	65
Figura 26 – Tamanho da filas após atuação do algoritmo de compartilhamento de banda . . . . .	66
Figura 27 – Total de pacotes enviados com sucesso. . . . .	67
Figura 28 – Histograma pacotes enviados com sucesso . . . . .	67
Figura 29 – Pacotes acumulados enviados com sucesso . . . . .	68
Figura 30 – Distribuição da quantidade total de pacotes enviados com erro . . . . .	69
Figura 31 – Histograma pacotes enviados com erro . . . . .	70
Figura 32 – Média do acúmulo de pacotes enviados com erro no tempo . . . . .	71
Figura 33 – Distribuição dos dados da vazão média . . . . .	72
Figura 34 – Histograma dos dados da vazão média . . . . .	72

# Lista de quadros

Quadro 1 – Elementos Topologia . . . . .	64
--	----

# Lista de tabelas

Tabela 1 – Questões de Pesquisa . . . . .	34
Tabela 2 – String de busca . . . . .	35
Tabela 3 – Bases de Busca . . . . .	35
Tabela 4 – Critérios de Inclusão . . . . .	36
Tabela 5 – Critérios de Exclusão . . . . .	36
Tabela 6 – Estudos Primários Selecionados . . . . .	37
Tabela 7 – Questões de pesquisa respondidas por trabalho. . . . .	42
Tabela 8 – Trabalhos relacionados . . . . .	49

# Lista de algoritmos

Algoritmo 1 – Monitoramento e aplicação de permuta de largura de banda . . . . .	59
--	----

# Lista de abreviaturas e siglas

API	Application Programming Interface
ATM	Asynchronous Transfer Mode
BCM	Bandwidth Constraint Model
BGP	Border Gateway Protocol
BSS	Bandwidth Sharing Service on Software-Defined Networks
CAB	CAching in Buckets
CORBA	Common Object Request Broker Architecture
CT	Class Type
DiffServ	Differentiated Services
DS-TE	DiffServ-Aware Traffic Engineering
DSCP	Differentiated Service Code Point
EDGE	Enhanced Data rates for GSM Evolution
FIN	Finally
GSM	Groupe Special Mobile
GSMP	General Switch Management Protocol
HSPA	High Speed Downlink Packet Access
HTTP	Hypertext Transfer Protocol
ICMPv6	Internet Control Message Protocol version 6
ID	Identity
IP	Internet Protocol
IPv6	Internet Protocol version 6
ISIS	Intermediate System to Intermediate System
LACP	Link Aggregation Control Protocol
LDP	Label Distribution Protocol

LTE	Long Term Evolution
MAC	Media Access Control
MAM	Maximum Allocation Model
MPLS	Multiprotocol Label Switching
NGN	Next Generation Networks
ONF	Open Networking Foundation
OSPF	Open Shortest Path First
OvS	Open vSwitch
OVSDB	Open vSwitch Database Management Protocol
PCE	Path Computation Element
PPP	Point-to-Point Protocol
QoE	Quality of Experience
QoS	Quality of Service
RDM	Russian Doll Model
REST	Representational State Transfer
RFC	Request for Comments
RPC	Remote Procedure Calling Protocol
RSVP	Resource Reservation Protocol
SDN	Software-Defined Networking
SLA	Service-Level Agreements
SOAP	Simple Object Access Protocol
SRAM	Static Random Access Memory
SYN	Synchronize
TCAM	Ternary Content-Addressable Memory
TCP	Transmission Control Protocol
IT	Information Technology

TLS	Transport Layer Security
TS	Type of Service
TTL	Time to Live
UDP	User Datagram Protocol
UMTS	Universal Mobile Telecommunication System
VLAN	Virtual Local Area Network
VM	Virtual Machine
VoIP	Voice over Internet Protocol

# Sumário

<b>1</b>	<b>Introdução</b>	<b>17</b>
1.1	Motivação	18
1.2	Hipótese	19
1.3	Objetivos	19
1.4	Metodologia	19
1.4.1	Classificação da Pesquisa	21
1.5	Organização	22
<b>2</b>	<b>Fundamentação teórica</b>	<b>23</b>
2.1	Redes Definidas por Software	23
2.2	QoS ( <i>Quality of Service</i> )	26
2.3	Modelos de Alocação de Banda	27
2.3.1	Maximum Allocation Model - (MAM)	28
2.3.2	Maximum Allocation with Reservation - (MAR).	28
2.3.3	Russian Doll Model - (RDM)	29
2.3.4	MARS – MAR <i>with Shapley value</i>	29
2.3.5	SHARM – SHApley <i>value with Reservation Model</i>	31
<b>3</b>	<b>Revisão Sistemática</b>	<b>33</b>
3.1	Planejamento	34
3.1.1	Questões de pesquisa	34
3.1.2	Estratégias de Busca	35
3.1.3	Bases de Busca	35
3.1.4	CrITÉrios de Inclusão e Exclusão	35
3.2	Execução	36
3.3	Análise	36
3.4	Resultados da Análise	39
3.4.1	Base dos trabalhos	39
3.4.2	Ano das Publicações	40
3.4.3	Citações das Publicações	41
3.5	Respostas às questões de pesquisa	41
3.5.1	QP1	42
3.5.2	QP2	43
3.5.3	QP3	44
3.6	Considerações Finais	44



<b>4</b>	<b>Trabalhos Relacionados</b>	<b>45</b>
4.1	Modelos de alocação de banda	45
4.2	Algoritmos heurísticos	46
4.3	Políticas de QoS	46
4.4	Estratégias de Alocação de Banda em SDN	47
4.5	Cosiderações Finais	48
<b>5</b>	<b>Modelo de Alocação de Banda com Critério de Justiça</b>	<b>50</b>
5.1	Conceitos de justiça em Redes de Computadores	50
5.2	MAMDFS - Maximum Allocation Model With Dynamic Fairness Sharing	51
5.3	Regras de alocação e compartilhamento	51
5.4	A arquitetura <i>BW-Manager</i>	54
5.5	Implementação do modelo MAMDFS	56
<b>6</b>	<b>Experimentação</b>	<b>60</b>
6.1	Metodologia Experimental	60
6.2	Planejamento experimental	61
6.2.1	Objetivo do experimento	61
6.2.2	Seleção de contexto	61
6.2.3	Variáveis	61
6.2.4	Formulação de Hipóteses	61
6.2.5	Seleção de Objetos	62
6.3	Design Experimental	62
6.4	Instrumentação	63
6.5	Operação do Experimento	63
6.5.1	Execução	63
6.5.2	Validação dos Dados	65
<b>7</b>	<b>Conclusão</b>	<b>73</b>
7.1	Contribuições	74
7.2	Dificuldades e limitações	74
7.3	Trabalhos futuros	74
	<b>Referências</b>	<b>76</b>

# 1

## Introdução

As redes de computadores atuais, compostas por diversos elementos de hardware e software, executam inúmeras aplicações que não foram projetadas, talvez sequer imaginadas, na época de seu projeto inicial no final da década de 1960. No intuito de suportar o crescente número de usuários e aplicações ao longo dos anos, diversos recursos foram adicionados. Porém, o seu núcleo se manteve inalterado. Esta arquitetura usa uma única infraestrutura para realizar o transporte de voz, dados e multimídia, por essa característica são denominadas de redes multisserviços.

Uma das características mais importantes em redes multisserviços é a alocação de recursos limitados em um cenário de alta demanda. Alocação de recursos trata-se de processos executados por elementos da rede para atender às demandas das aplicações ([Kim, 2013](#)). Os mecanismos de alocação devem ser capazes de atribuir de maneira eficiente os recursos disponíveis entre as diversas aplicações, otimizando seu uso.

Outra característica que impõe grandes limitações a este tipo de infraestrutura, é o fato de que os switches, roteadores e demais elementos da rede não podem ter seus *firmwares* reprogramados e novos protocolos não podem ser testados em um ambiente de produção. Essas características dificultam a alteração e impossibilitam a evolução da arquitetura das redes atuais, fato esse classificado por alguns pesquisadores como ossificação da rede ([McKeown et al., 2008](#)). A partir desse contexto, surge a ideia de redes definidas por software como uma forma de evolução das redes de computadores ([Nunes et al., 2014](#)).

Tanto a comunidade acadêmica quanto os principais fabricantes de equipamentos de rede têm apresentado uma grande aceitação da arquitetura SDN, uma vez que de forma crescente soluções voltadas para esta arquitetura estão sendo desenvolvidas ([Cisco, 2020](#)). Porém, o conceito de redes programáveis é algo que vem amadurecendo ao longo de décadas. Em meados dos anos 1990 até o início dos anos 2000, foram introduzidas funções programáveis nas redes ativas com o intuito de permitir uma maior inovação. Em seguida, entre os anos de 2001 e 2007, foi

implementada a separação dos planos de controle e de dados, juntamente com o desenvolvimento de interfaces abertas para a comunicação entre os planos. Entre os anos de 2007 e 2010, foram criados os sistemas operacionais de redes e o protocolo *OpenFlow*, que representa a primeira instância da adoção generalizada de uma interface aberta tornando prática e escalável a separação dos planos de dados e controle (Feamster et al., 2014).

Este capítulo introduz o contexto no qual se aplicam os conceitos de Compartilhamento de Banda e Redes SDN. Em seguida, apresenta o objeto de estudo, a estratégia adotada e a organização deste trabalho.

## 1.1 Motivação

A grande quantidade de serviços e aplicativos executados atualmente nas redes de computadores, tais quais as mídias sociais, computação em nuvem, dispositivos móveis, serviços de VoIP (*Voice over IP*), IoT (*Internet of Things*) e uma infinidade de novos produtos com capacidade de conectar-se à Internet estão cada vez mais forçando as redes convencionais para seus limites. Além disso, o grande fluxo de informações demanda a implementação de técnicas de otimização de recursos, como aplicação de políticas de QoS (*Quality of Service*) distintas para cada tipo de tráfego, acarretando em um aumento no nível de complexidade de gerenciamento e elevando também os custos de manutenção da infraestrutura.

A complexidade da arquitetura das redes tradicionais demanda a utilização de componentes que encapsulam tecnologias diferentes, obrigando os administradores se manterem em constante especialização em diversos equipamentos fornecidos por fabricantes distintos, resultando em menos agilidade nos processos e mais custos operacionais.

Esta complexidade também exige cada vez mais disponibilidade de recursos de rede. Diante deste contexto, surgem dificuldades associadas à capacidade que as operadoras têm em oferecer serviços que suportem os diferentes tipos de fluxo, tornando cada vez mais custosa, tanto de uma perspectiva econômica quanto operacional, a tarefa de prover uma arquitetura capaz de acomodar os mais diversos tipos de serviços existentes.

Vários estudos, a exemplo dos apresentados em (Awduche et al., 1999), (Kurose and Ross, 2010), (Salgueiro, 2009), (Suliman et al., 2004), (Zhang et al., 2009), discutem como a utilização de estratégias e mecanismos de compartilhamento de recursos em redes podem alcançar certos benefícios, tais como um gerenciamento mais simplificado e otimização do uso dos recursos da rede.

A arquitetura SDN surge como um contraponto ao padrão tradicional das redes de computadores, proporcionando um gerenciamento mais simplificado graças à separação de camadas, centralização do controle e utilização de padrões e protocolos abertos. Estas características da arquitetura SDN permitem com mais facilidade a extensão das funcionalidades

da camada de controle, incluindo mecanismos que auxiliam no processo de gerenciamento de recursos disponíveis, visando um melhor aproveitamento e minimizando custos, principalmente operacionais.

Diante deste contexto, este trabalho apresenta uma arquitetura que possibilita o compartilhamento de largura de banda implementando modelos de alocação de banda com justiça em um controlador SDN, contribuindo para um melhor aproveitamento dos recursos disponíveis de forma dinâmica e eficiente.

## 1.2 Hipótese

Um controlador SDN pode prover um mecanismo de compartilhamento de largura de banda com justiça através da implementação e utilização de modelos de alocação de banda, otimizando a utilização de recursos e melhorando o desempenho em redes definidas por software.

## 1.3 Objetivos

O objetivo geral desta dissertação é desenvolver uma arquitetura na qual são implementados modelos de alocação de recursos seguindo critérios de justiça em controladores de redes definidas por software, com o intuito de gerenciar o compartilhamento de banda entre classes de tráfego distintas. Para atingir o objetivo geral, foram estabelecidos os seguintes objetivos específicos:

- Desenvolver uma arquitetura que, implementando modelos de alocação de banda com justiça, ajude na otimização da utilização dos recursos de uma rede com enlace compartilhado;
- Implementar modelos de alocação de banda com justiça;
- Desenvolver um módulo em um controlador SDN para a aplicação da solução fornecida pela arquitetura;

## 1.4 Metodologia

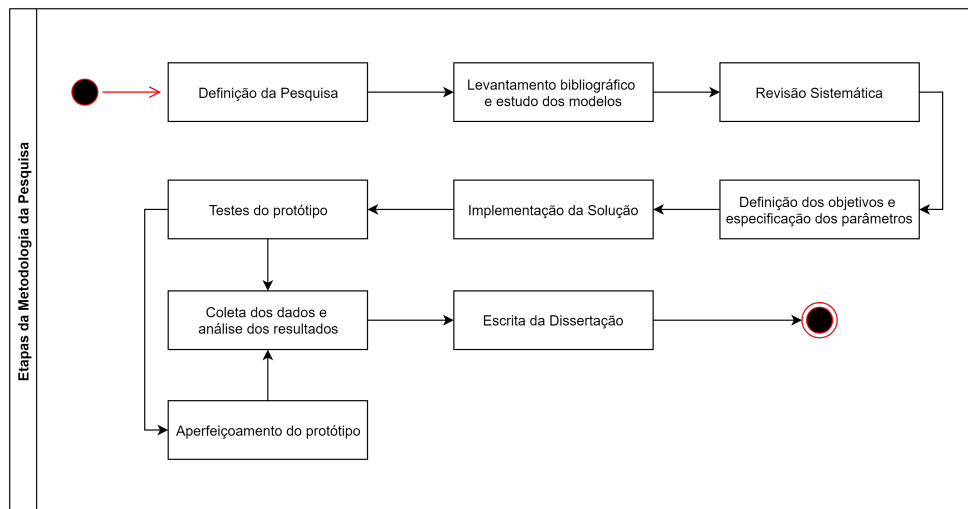
Uma metodologia tem como objetivo identificar os processos e atividades a serem realizadas para estudar e resolver um determinado problema. A fim de auxiliar o processo de desenvolvimento, foram definidas as etapas ilustrados na Figura 1. Após a definição das etapas, foram desenvolvidas atividades de forma sequencial, as quais são descritas mais detalhadamente a seguir:

1. **Definição da Pesquisa:** o interesse em redes Redes Definidas por Software surgiu após participar, ainda durante a graduação, de um projeto de pesquisa nesta área e perceber o

seu grande potencial exploratório. Por se tratar, principalmente, de uma área relativamente nova, modelos de alocação de banda em SDN ainda não são amplamente utilizados para prover um melhor aproveitamento de recursos em termos de redes de computadores.

2. **Levantamento bibliográfico e estudo dos modelos:** Após a definição do tema da pesquisa, foi realizado um estudo bibliográfico na literatura com o intuito de melhor compreender os modelos de alocação de banda, SDN e conceitos de justiça em redes de computadores.
3. **Revisão Sistemática:** nessa fase, foram realizadas buscas nas bases de artigos científicos com o propósito de identificar na literatura trabalhos existentes que abordam o tema, bem como analisar os trabalhos relacionados resultantes da revisão sistemática.
4. **Definição dos objetivos e especificação dos parâmetros:** após a realização da revisão sistemática, foram definidos os objetivos do trabalho e especificação dos parâmetros (cargas de trabalho, métricas a serem analisadas, protocolos de rede utilizados, etc.).
5. **Implementação da Solução:** nessa etapa, deu-se início a construção de um software protótipo no qual foram implementados algoritmos que tornaram possível aplicar as técnicas de alocação de banda em redes definidas por software.
6. **Testes do protótipo:** nessa fase, foram realizados os primeiros testes do software protótipo, depurações e geração de dados para análises.
7. **Coleta dos dados e análise dos resultados:** nessa etapa foram coletados os dados gerados na fase de testes do protótipo. Após uma análise dos dados, foram detectadas algumas inconsistências nas informações, o que revelou a necessidade de correções e um aperfeiçoamento do software.
8. **Aperfeiçoamento do protótipo:** essa atividade consistiu em realizar algumas correções no código-fonte do software para resolver problemas relacionados ao mecanismo de alocação de largura de banda e também em relação à consistência das informações.
9. **Escrita da Dissertação:** nessa última etapa do trabalho, foi realizado o processo de escrita da dissertação após validação dos dados coletados através de métodos estatísticos.

Figura 1 – Etapas da Metodologia da Pesquisa



Fonte: (Autor, 2020)

### 1.4.1 Classificação da Pesquisa

De acordo com (Nascimento and Souza, 2017), uma metodologia pode ser classificada de acordo com sua natureza, seus métodos, objetivos e procedimentos. Sendo assim, este trabalho foi realizado a partir de uma perspectiva metodológica de natureza aplicada, já que a pesquisa aplicada tem como objetivo gerar conhecimento para resolução de problemas específicos.

Do ponto de vista metodológico, este trabalho pode ser classificado como uma pesquisa quantitativa, uma vez que emprega medidas padronizadas para comparar e analisar estatisticamente os dados (Nascimento and Souza, 2017).

Nesse trabalho foi realizada uma pesquisa bibliográfica e experimental. Bibliográfica por utilizar informações provenientes de bases de artigos científicos, livros, sítios da internet e outras ferramentas com acesso público (Marconi and Lakatos, 2003). Experimental por fazer uso de provas de conceito para validar a solução proposta, investigando e analisando fenômenos em um determinado ambiente (Yin, 2014). Sendo assim, em relação aos procedimentos, este trabalho pode ser classificado como bibliográfico e experimental.

Em relação aos objetivos, pode-se classificar este trabalho como uma pesquisa exploratória, uma vez que busca um maior entendimento do problema para torná-lo explícito, para aprimorar ideias ou descobrir intuições (Gil, 2008).

## **1.5 Organização**

No Capítulo 2 é apresentada a fundamentação teórica introduzindo os conceitos de SDN, modelos de alocação de banda e uma breve introdução à Teoria dos Jogos Cooperativos. No Capítulo 3 é apresentada uma revisão sistemática do estado da arte, seguido pelo Capítulo 4 com os trabalhos relacionados. Os Capítulos 5, 6 e 7 apresentam, respectivamente, o modelo de alocação de banda com critérios de justiça proposto neste trabalho, os experimentos para validação da solução e a conclusão.

# 2

## Fundamentação teórica

Neste Capítulo são apresentados os conceitos de SDN, Qualidade de Serviço (QoS), modelos de alocação de banda clássicos e dois novos modelos de alocação de recursos, onde são aplicados os conceitos da Teoria dos Jogos Cooperativos para o cálculo de alocação de largura de banda com justiça.

### 2.1 Redes Definidas por Software

A arquitetura SDN é um novo conceito em redes de computadores onde sua principal característica é o desacoplamento das camadas de dados, controle e infraestrutura. Essa segregação é a principal diferença em termos de arquitetura quando comparado às redes tradicionais.

As redes concebidas a partir dessa nova abordagem possuem também uma camada de controle centralizada e totalmente programável.

Como há separação dos planos de controle e de dados, é crucial que a comunicação entre estes elementos seja padronizada. Por essa razão, a *Open Networking Foundation* (ONF) propôs a utilização de um protocolo que especifica a utilização de uma interface de comunicação em SDN.

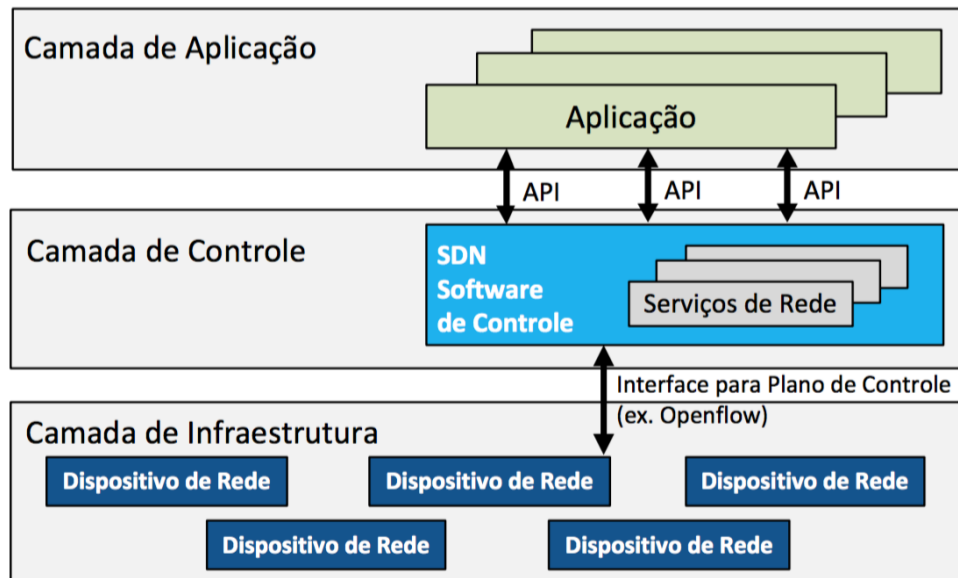
Segundo (ONF, 2013), alguns objetivos principais da arquitetura SDN são:

- Unificar o gerenciamento e controle dos dispositivos de rede independentemente do seu fabricante;
- Definir uma interface de comunicação aberta e padronizada para a gerência via aplicação;
- Fomentar o avanço e a inovação no desenvolvimento de novas aplicações;
- Possibilitar que operadores de redes possam programar a rede em tempo real via aplicação;



A Figura 2 ilustra de maneira mais detalhada a arquitetura de uma rede definida por software.

Figura 2 – Detalhes Arquitetura SDN

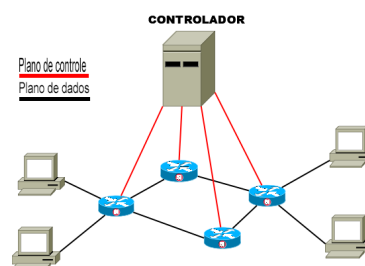


Fonte: Adaptado de (SDX, 2018).

A principal camada em uma rede com arquitetura SDN é a camada de controle. Esta camada abriga o controlador de rede, que corresponde a um software responsável por gerenciar o controle de fluxo para os *switches*/roteadores, aplicativos e lógica de negócios que implementam as redes inteligentes (Sdx-Central, 2018).

Controladores SDN geralmente contêm uma coleção de módulos “plugáveis” que podem executar diferentes tarefas de rede. Algumas das tarefas básicas incluem o inventário de quais dispositivos estão dentro da rede e os recursos de cada um, reunindo estatísticas de rede, etc. Extensões podem ser inseridas para melhorar a funcionalidade e suportar recursos mais avançados, como executar algoritmos capazes de analisar e orquestrar novas regras em toda a rede. A Figura 3 ilustra uma visão geral sobre a atuação de um controlador sobre uma arquitetura SDN, onde as linhas em vermelho representam o plano de controle e as linhas em preto o plano de dados.

Figura 3 – Controlador SDN

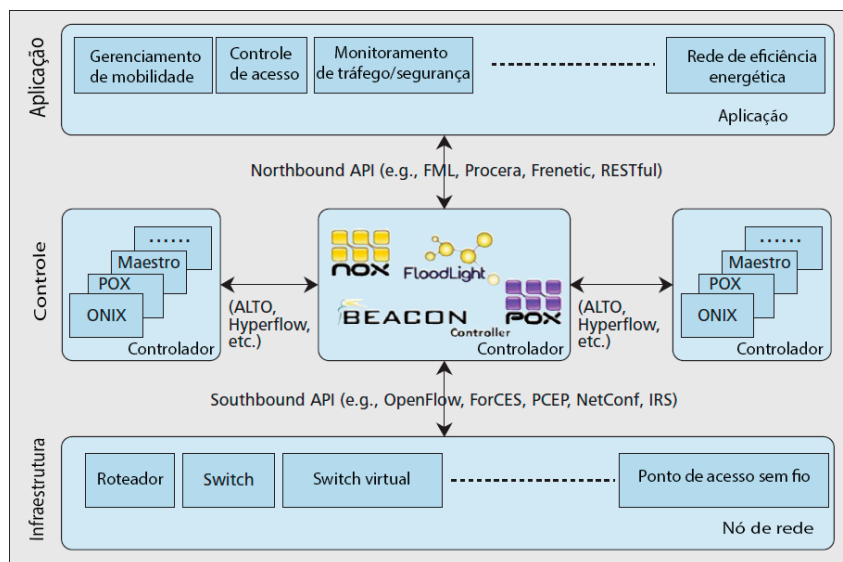


Fonte: (Autor, 2020)

A comunicação entre os controladores e os dispositivos de rede, bem como os aplicativos de regra de negócio, é realizada através das APIs *Southbound* e *Northbound*, respectivamente. Estas APIs são formadas por conjuntos de interfaces que possibilitam a comunicação entre as camadas da arquitetura SDN.

A Figura 4 ilustra a arquitetura de um controlador de redes SDN, onde a comunicação entre as camadas de aplicação e de controle se dá principalmente através de uma RESTful API fornecida pelo próprio controlador, além da possibilidade de utilização de outras interfaces programáticas. A comunicação entre a camada de controle e de infraestrutura é realizada através da utilização de diversos protocolos, entre esses está o protocolo *OpenFlow*. Observa-se também que mais de um controlador pode ser utilizado na camada de controle, os quais se comunicam entre si utilizando diversos protocolos de rede.

Figura 4 – Arquitetura do Controlador SDN.



Fonte: Adaptado de (Sezer et al., 2013).

Projetado originalmente pela Universidade Stanford como um padrão aberto, o protocolo *OpenFlow* consiste em uma tecnologia proposta para padronizar a maneira como o controlador SDN se comunica com os dispositivos da camada de infraestrutura, definindo também um protocolo para essa comunicação. Sendo assim, o protocolo fornece meios de controlar os dispositivos de rede compatíveis com esta tecnologia sem a necessidade de os fabricantes exporem o código de seus produtos (Lara and Ramamurthy, 2014). Deste modo, diferentemente das redes de computadores tradicionais, os controladores podem usar este protocolo para configurar de forma automática as tabelas de fluxos dos *switches*, sem a necessidade de intervenção humana.

As tabelas de fluxo são utilizadas pelos *switches* que implementam o protocolo *OpenFlow* para classificação dos fluxos e as ações que deverão ser tomadas para o cada um. É na tabela de fluxos que são definidos também os níveis de prioridade de cada fluxo.

Outro elemento de grande importância no processo de comunicação entre as camadas de controle e infraestrutura é o *Open vSwitch* (Open Virtual Switch - OVS). O *Open vSwitch* é um projeto de software de código aberto projetado para ser usado como um switch virtual em ambientes de rede virtualizados. É extensível programaticamente e seu controle é feito através do *OpenFlow* e pelo protocolo de gerenciamento OVSDB (Open vSwitch Database) ([RFC-7047, 2013](#)).

O protocolo OVSDB é usado pelo *Open vSwitch* principalmente para monitoramento e configuração das regras de encaminhamento. O OVSDB é um protocolo de gerenciamento moderno utilizado para configurar o OVS, o que significa criar, excluir e modificar *bridges*, portas, filas e interfaces.

Já o processo de comunicação entre o controlador de rede e aplicativos de regras de negócio, é realizada de forma menos complexa, sendo na maioria dos casos, através de APIs REST oferecidas pelos controladores.

O uso de controladores SDN permite programar o comportamento da rede por meio de softwares, adicionando controle e gerenciamento de recursos de rede de forma inteligente e automatizada, além de permitir também o avanço no desenvolvimento de novas técnicas e serviços.

## 2.2 QoS (*Quality of Service*)

QoS (*Quality of Service*) é a capacidade de melhorar os serviços trafegados na rede e tem como objetivo fornecer serviços de rede melhores e mais previsíveis, fornecendo largura de banda dedicada, *jitter* controlado e latência, e perda de características melhoradas. QoS oferece serviços de rede inteligente que, quando corretamente aplicadas, ajudam a fornecer desempenho consistente e previsível ([Cisco Systems, 2009](#)).

A arquitetura de redes de computadores atuais foi projetada para o envio de informações utilizando o modelo do melhor esforço (*Best Effort*). Nesse modelo a transmissão ocorre sem nenhum critério de priorização, sem quaisquer garantias de vazão, atraso ou outro requisito de QoS. O requisito de banda destinado a cada fluxo é o maior possível enquanto não houver nenhum tipo de congestionamento. Havendo congestionamento, ocorre o descarte de pacotes e, conseqüentemente, queda na taxa de transmissão.

Existem dois princípios básicos para a aplicação de políticas de QoS em redes de computadores:

- *Intserv*: Serviços integrados;
- *Diffserv*: Serviços diferenciados.

*Intserv* é uma arquitetura de QoS, que tem o propósito de garantir níveis de qualidade de serviço ponta a ponta, através de reserva de recursos e estabelecimento de chamada. Essa arquitetura utiliza principalmente o protocolo de reservas de recursos RSVP (*Resource Reservation Protocol*), descrito na RFC 2205, para sinalizar as necessidades de QoS para cada dispositivo ao longo da rede, possibilitando vários transmissores enviarem os dados para vários grupos de receptores, eliminando o congestionamento da rede.

O RSVP permite que vários transmissores enviem os dados para vários grupos de receptores, torna possível receptores individuais mudarem livremente de canais e otimiza o uso da largura de banda ao mesmo tempo que elimina o congestionamento (Tanenbaum, 2003).

*DiffServ* são serviços baseados no tratamento de classes de maneira diferenciada, podendo lidar com tipos de classes diferentes dentro de uma rede. O tratamento diferenciado é replicado em todos nós da rede, deste modo, os pacotes de dados marcados como prioritários, ao chegarem em um roteador são separados e tratados de forma diferenciada. Sendo assim, *DiffServ* permite a aplicação de QoS através da diferenciação de classes de serviços. A RFC 4594 descreve o processo de mapeamento de aplicações para serviço.

A principal tecnologia capaz de oferecer serviços diferenciados é a MPLS (*Multi-Protocol Label Switching*). Essa tecnologia é utilizada em backbones com o objetivo de solucionar os problemas de velocidade, escalabilidade e gerenciamento de qualidade de serviço. Funciona rotulando os pacotes conforme vão entrando e saindo da rede, processando e classificando os dados em classes de serviço, e em seguida enviando os dados através de caminhos que foram determinados pelas classes. Os caminhos pelos quais os pacotes irão percorrer até seu destino em uma rede MPLS, é denominado LSP (*Label Switch Path*).

Em SDN, as políticas de QoS são implementadas através de filas. As filas são projetadas para fornecer uma garantia sobre a taxa de fluxo dos pacotes de dados colocados na fila. Dessa forma, diferentes filas com taxas diferentes podem ser utilizadas para priorizar tráfegos considerados mais importantes ou até mesmo limitar algum tráfego não prioritário.

As filas são criadas nas portas dos *switches* através de comandos executados pelo protocolo OVSDB, por exemplo, e o protocolo OpenFlow fornece o mecanismo de enfileiramento informando o controlador da existência dessas filas.

## 2.3 Modelos de Alocação de Banda

Em redes DS-TE (*DiffServ aware Traffic Engineering*), a alocação de largura de banda entre várias classes é determinada a partir da relação entre o tipo de classe (*Class-Type* - CT) e a restrição de largura de banda (*Bandwidth Constraint* - BC) (Adami et al., 2008). CT está definido na RFC 5455 como um grupo de troncos de tráfego com base em seus valores de QoS para que eles compartilhem a mesma reserva de largura de banda. Um único tipo de classe pode representar

uma ou mais classes. CT é usado para alocação de largura de banda, roteamento de restrição e controle de admissão. A restrição de largura de banda (BC) é um limite na porcentagem de largura de banda de um link que um determinado tipo de classe pode usar. No contexto das redes DS-TE, modelos de alocação de banda são de extrema importância para alcançar uma utilização eficiente dos recursos da rede. Determinam como a largura de banda do link é dividida entre várias Classes de Tráfego e define a relação entre CT e BC.

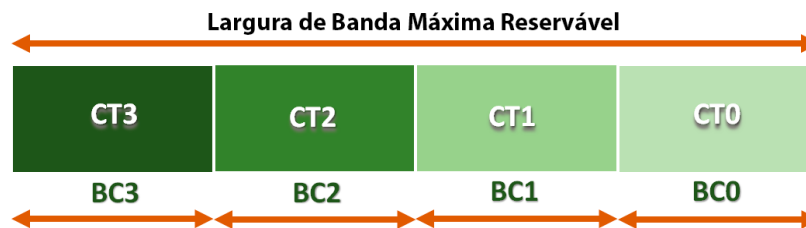
As subseções seguintes apresentam os conceitos dos modelos de alocação de banda clássicos e também dois novos modelos propostos na literatura, que têm como objetivo a alocação de recursos com justiça.

### 2.3.1 Maximum Allocation Model - (MAM)

Em (Faucheur and Lai, 2005) foi especificado o primeiro e o mais simples modelo de restrição de largura de banda que objetiva realizar o mapeamento de um BC em um tipo de classe. Desta forma, a largura de banda em cada um CT é separada de outros CTs.

Na literatura, o exemplo apresentado em (Lai, 2005) é amplamente adotado e referenciado como uma representação canônica de um BCM (*Bandwidth Constraint Model*). Neste exemplo, é utilizado um índice para identificar uma BC que corresponde a uma CT. Sendo assim,  $CT_i$  corresponde ao  $i$ -ésimo tipo de classe e  $BC_i$  à restrição de banda do tipo de classe  $i$ . A Figura 5 exibe a representação canônica do MAM com 4 CTs.

Figura 5 – Maximum Allocation Model - (MAM)



Fonte: Adaptado de (Faucheur and Lai, 2005)

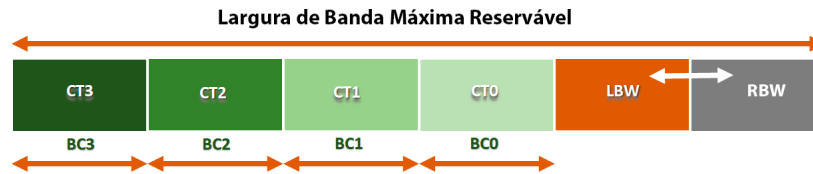
O MAM tem como vantagem a capacidade de garantir que a reserva da largura de banda não seja compartilhada entre outros CTs na faixa de seu BC. Todavia, sua grande desvantagem é o fato de que um CT não pode fazer uso da largura de banda deixada sem uso por outro CT.

### 2.3.2 Maximum Allocation with Reservation - (MAR).

O modelo MAR, (Ash, 2005), é uma extensão do MAM, mas com complexidades adicionais, pois a diferença entre eles é que os CTs podem exceder seu BC se não houver congestionamento nos outros. Caso haja o congestionamento, os CTs voltam à largura de banda previamente alocada.

Para o exemplo mostrado na Figura 6, a banda máxima para cada classe é calculada de acordo com uma dada carga  $L_i$  da classe  $i$ . São definidas três prioridades para acesso à banda compartilhada: normal, média e alta. Uma classe com prioridade normal tem sua BC calculada em proporção à carga  $L_i$  e limitada por uma banda máxima reservável **RBW** que pode ser compartilhada por todas os CTs (Salgueiro, 2009).

Figura 6 – Maximum Allocation with Reservation - (MAR)

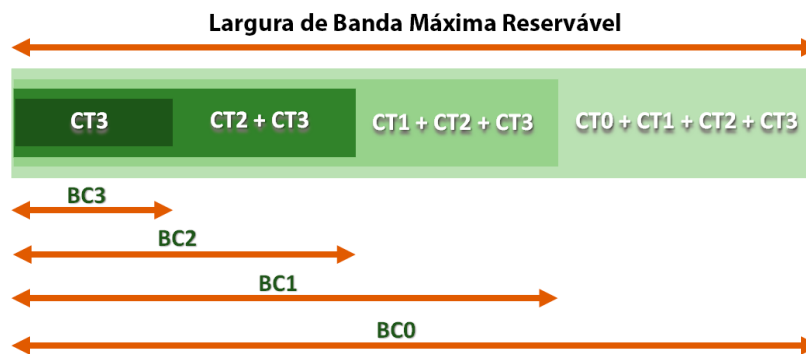


Fonte: Adaptado de (Ash, 2005)

### 2.3.3 Russian Doll Model - (RDM)

Especificado por (Faucheur, 2005), o *Bandwidth Constraint RDM* difere do MAM ou MAR por permitir que as CTs compartilhem a largura de banda entre si. Ele atribui BCs a grupos de CTs. Por exemplo, CT0 com os mais altos requisitos de QoS obtém seu próprio BC0. O CT1 com requisitos de QoS mais baixos compartilha seu BC1 com o CT0 e assim por diante. Em casos extremos, as prioridades mais baixas obtêm menos largura de banda do que precisam. A fim de simplificação, a Figura 7 mostra como o RDM funciona com 4 CTs.

Figura 7 – Russian Doll Model - (RDM)



Fonte: Adaptado de (Faucheur, 2005)

### 2.3.4 MARS – MAR with Shapley value

Nesta Seção o modelo de alocação de recursos MARS - MAR with Shapley value é apresentado. Este modelo utiliza o Valor de Shapley na formulação de suas regras de alocação,

portanto, faz-se necessária uma contextualização sobre a Teoria dos Jogos Cooperativos.

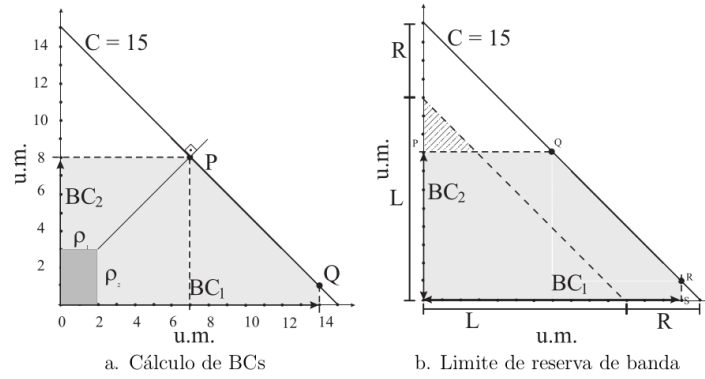
A Teoria dos Jogos teve seu desenvolvimento iniciado por von Neumann, em 1928, com a demonstração do teorema *minimax*. Porém, somente após firmar parceria com Morgenstern, em 1944, que a atenção do público foi despertada pela proposta de análise do comportamento econômico através da perspectiva do "jogo de estratégia", criando uma expectativa de reformulação da teoria econômica em uma base totalmente nova, na qual o conceito de "processo competitivo" seria reestruturado em termos de mecanismos onde os agentes econômicos atuam estrategicamente (von NEUMANN and MORGENSTERN, 1972).

Em 1950, John F. Nash demonstrou o teorema minimax para grandes números de agentes. Em 1952, Lloyd Shapley apresenta o conceito de "núcleo".

Em redes de computadores, os primeiros trabalhos sobre Teoria dos Jogos foram realizados sob uma perspectiva econômica, onde eram discutidas políticas de preços na Internet considerando os requisitos das aplicações. Algumas das estratégias incentivavam a utilização de recursos subutilizados e penalizava o uso de recursos com maior demanda. Recentemente tem crescido o interesse em aplicar os conceitos da Teoria dos Jogos em redes de computadores para a modelagem de soluções para problemas controle de congestionamento, roteamento, alocação de banda, entre outros.

O modelo MARS – *MAR with Shapley value* (Salgueiro, 2009), da mesma forma que o modelo MAR, utiliza especificações de pesos e reservas de recursos em sua regra de alocação. A principal diferença entre estes dois modelos é que no caso do MARS, a regra de divisão proporcional é substituída pelo Valor de Shapley. Os pesos definidos para o MAR são preservados. A regra de divisão é baseada na contribuição individual de cada possível coalizão especificada no jogo. Inicialmente todos os CTs são considerados com direitos iguais. Após o cálculo das parcelas de recurso, pesos são utilizados para associar prioridades de acesso aos CTs. A regra de alocação do MARS foi definida com o objetivo de comparar o impacto da substituição do particionamento proporcional pelo valor de Shapley. A Figura 8 ilustra a regra de alocação do MARS, onde os conjuntos de pontos entre as retas PQ e QR representam os conjuntos de possíveis alocações. A variável R, que representa o valor que pode ser reservado em períodos de sobrecarga, equivale a 4 e indica que o CT<sub>2</sub> pode exceder o limite do BC<sub>2</sub> até  $L = C - R = 11$ , até a soma da capacidade total alocada for menor que  $C - R$ .

Figura 8 – Regra de alocação MARS



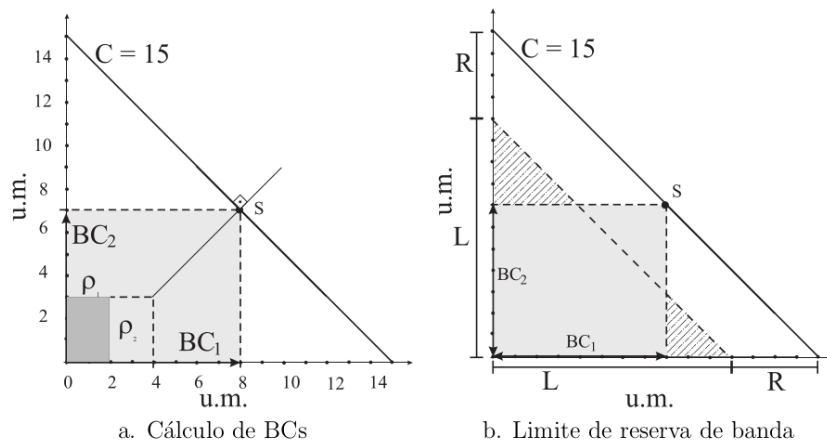
Fonte: (Salgueiro, 2009)

### 2.3.5 SHARM – SHAPley value with Reservation Model

O modelo SHARM – *SHARM – SHAPley value with Reservation Model* (Salgueiro, 2009), assim como o MARS, utiliza o Valor de Shapley para formular suas regras de alocação, incluindo pesos como parte de sua definição. Pesos representam um meio de aumentar a parcela de banda para os CTs com prioridades mais altas. Este tratamento diferenciado garante que a QoS seja mantida em períodos de sobrecarga ao menos para os CTs com prioridade alta.

Pode-se observar na Figura 9 a regra de alocação do SHARM, onde os BCs são calculados diretamente a partir do Valor de Shapley. A relação entre os BCs e o parâmetro de intervalo de reserva  $R$  que pode ser compartilhado por todos os CTs. Nesta figura,  $R = 4$  significa que tanto o CT 1 quanto o CT 2 podem extrapolar seus limites, enquanto a soma da capacidade alocada total for menor do que o recurso  $C - R$ .

Figura 9 – Regra de alocação SHARM



Fonte: (Salgueiro, 2009)

Neste Capítulo foram abordados os principais conceitos e características dos elementos que compõem uma arquitetura SDN. Foram apresentados também, conceitos relacionados a



modelos de alocação de banda e QoS.

Para garantir a QoS, os serviços diferenciados (*Diffserv*) operam com base em diferentes classes de tráfego com o objetivo de melhorar o desempenho da rede aplicando diferentes restrições de largura de banda em diferentes classes. Modelos de alocação de largura de banda atuam com o propósito de alocar recursos para classes de tráfego de forma individual. Diversos modelos podem ser utilizados para esse fim de acordo com as necessidades e políticas pré definidas.

Segundo [Karakus and Durresi \(2017\)](#), arquitetura SDN oferece recursos que podem beneficiar a implantação de políticas de QoS. Entre esses recursos destacam-se: o encaminhamento baseado em fluxo, permitindo o tratamento diferenciado para os fluxos de aplicativos. A atualização dinâmica de regras de fluxo que permite atualização imediata das regras instaladas nos dispositivos de rede sem a necessidade de interromper o seu funcionamento. O controle centralizado fornece uma visão geral da rede e possibilita o monitoramento e análise dos fluxos em tempo real. Além disso, o protocolo *OpenFlow* possibilita o monitoramento e gerenciamento de filas trabalhando em conjunto com outros protocolos, como o *OVSD*.

Com o objetivo de identificar trabalhos relacionados a aplicação de políticas de QoS e modelos de alocação de largura de banda em ambientes SDN, foi realizada uma Revisão Sistemática da Literatura. O próximo Capítulo apresenta todos os passos seguidos durante o processo de revisão, inclusive a análise dos principais estudos resultantes.

# 3

## Revisão Sistemática

A Revisão Sistemática da Literatura (RSL) é um processo de busca no qual faz-se uso de um protocolo bem definido, de forma que outros pesquisadores, ao seguir os mesmos passos definidos pela revisão sistemática, sejam capazes de obter os mesmos resultados.

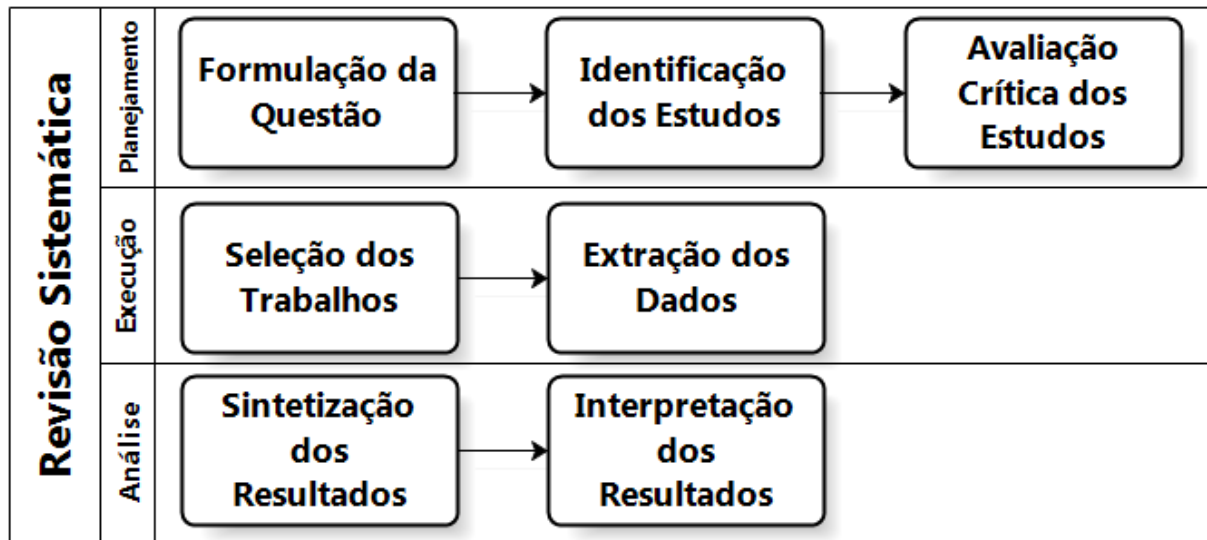
De acordo com ([Petersen et al., 2008](#)), uma revisão sistemática consiste em definir questões de pesquisa que abrangem o escopo de interesse da revisão, realizar buscas em bases de artigos científicos para obtenção dos estudos primários e filtrar os estudos com base no tema de interesse. Diante dos resultados das buscas, faz-se necessário a realização de uma seleção dos artigos relevantes às questões de pesquisa definidas, em seguida uma leitura cuidadosa para a obtenção das respostas para as questões de pesquisa e a apresentação das análises dos resultados obtidos. Este capítulo descreve como foi realizado o método da busca, seleção dos artigos e quais critérios foram utilizados na filtragem dos artigos.

Uma RSL possibilita a identificação, avaliação e interpretação dos trabalhos disponíveis relevantes para determinadas questões de pesquisa. Estudos individuais que contribuem para uma revisão sistemática são chamados estudos primários.

A condução da uma revisão sistemática deve ser realizada seguindo uma estratégia de pesquisa pré-definida, que permita avaliar a integridade da pesquisa e identificar quais apoiam ou não as suas hipóteses ([Kitchenham, 2004](#)).

Pode-se observar na Figura 10, que a RSL é constituída das fases de planejamento, execução e análise. Na fase de planejamento as questões de pesquisa são elaboradas para análise dos estudos primários, a seleção das bases onde serão realizadas as buscas e a definição dos termos de busca que serão utilizadas. Na fase de execução, são feitas as buscas utilizando as strings de busca, além disso, através da aplicação dos critérios de inclusão e exclusão, a filtragem dos estudos primários é realizada. Na fase de análise, os dados são sintetizados para responder as questões de pesquisa juntamente com a produção de um resumo dos trabalhos selecionados.

Figura 10 – Fases de uma revisão sistemática



Fonte: (Couto, 2013)

### 3.1 Planejamento

Nesta fase é criado o protocolo do mapeamento sistemático, a definição dos objetivos do trabalho, as hipóteses de pesquisa, as questões de pesquisa, as bases de buscas que serão utilizadas, as strings de busca que serão aplicadas na busca e os critérios de inclusão e exclusão.

#### 3.1.1 Questões de pesquisa

O primeiro passo foi definir a área de interesse da pesquisa. O tema escolhido foi Modelos de Alocação de Banda em Redes Definidas por Software. Para nortear os estudos foram definidas algumas questões de pesquisa, tendo em vista que essas questões, uma vez respondidas, tornaram sólido o embasamento, fortaleceriam o conhecimento teórico e as características experimentais necessárias para validar as soluções.

Tabela 1 – Questões de Pesquisa

#	Questões de Pesquisa
QP1	Foram utilizadas ferramentas na validação da solução avaliada?
QP2	Quais métricas foram utilizadas para medir a qualidade da solução avaliada?
QP3	Algum modelo/estratégia de alocação de banda é implementado em um controlador SDN?

Fonte: (Autor, 2020)

### 3.1.2 Estratégias de Busca

Com base nas questões de pesquisas elaboradas foram definidas as palavras-chave para compor as strings de buscas. O escopo das buscas foi delimitado fazendo uso dos operadores lógicos AND e OR. A seguir, as strings de busca são apresentadas na Tabela 2.

Tabela 2 – String de busca

#	String
Inglês	("bandwidth allocation"AND sdn) OR ("bandwidth allocation model"AND sdn) OR ("bandwidth allocation models"AND "software-defined networking"))

Fonte: (Autor, 2020)

### 3.1.3 Bases de Busca

As bases de dados foram selecionadas de acordo com a relevância das mesmas. As buscas foram realizadas através dos portais disponibilizados pelas bases na Internet. Foram considerados apenas trabalhos publicados a partir de 2012, já que início do movimento para dissociar os planos de controle e encaminhamento para permitir a inovação partiu do ano de 2011. A Tabela 3 apresenta as bases escolhidas para as buscas.

Tabela 3 – Bases de Busca

Base	URL
Wiley Online Library	<a href="https://onlinelibrary.wiley.com/">https://onlinelibrary.wiley.com/</a>
Science Direct	<a href="https://www.sciencedirect.com/">https://www.sciencedirect.com/</a>
Scopus	<a href="https://www.scopus.com">https://www.scopus.com</a>
IEEE Xplore	<a href="http://ieeexplore.ieee.org">http://ieeexplore.ieee.org</a>
Springer Link	<a href="https://link.springer.com/">https://link.springer.com/</a>
ACM Digital Library	<a href="https://dl.acm.org/">https://dl.acm.org/</a>

Fonte: (Autor, 2020)

### 3.1.4 Critérios de Inclusão e Exclusão

Durante o processo de revisão sistemática, os critérios de inclusão e exclusão são aplicados para classificar os estudos primários e definir quais devem ou não ser selecionados para um estudo mais aprofundado nas etapas seguintes. As Tabelas 4 e 5 apresentam os critérios de inclusão e exclusão definidos nesta revisão sistemática.

Tabela 4 – Critérios de Inclusão

#	Critérios de Inclusão
1	Os estudos devem ser trabalhos completos.
2	Os estudos devem estar disponíveis na web.
3	Os estudos devem ter sido publicados em journals, simpósios ou conferências das bases citadas.

Fonte: (Autor, 2020)

Tabela 5 – Critérios de Exclusão

#	Critérios de Exclusão
1	Estudos resumidos.
2	Estudos não disponíveis através do sítio Periódicos Capes ou Web.
3	Estudos que não abordam modelos de alocação de banda em SDN
4	Estudos publicados mais de uma vez. Neste caso, os estudos mais completos serão selecionados.

Fonte: (Autor, 2020)

## 3.2 Execução

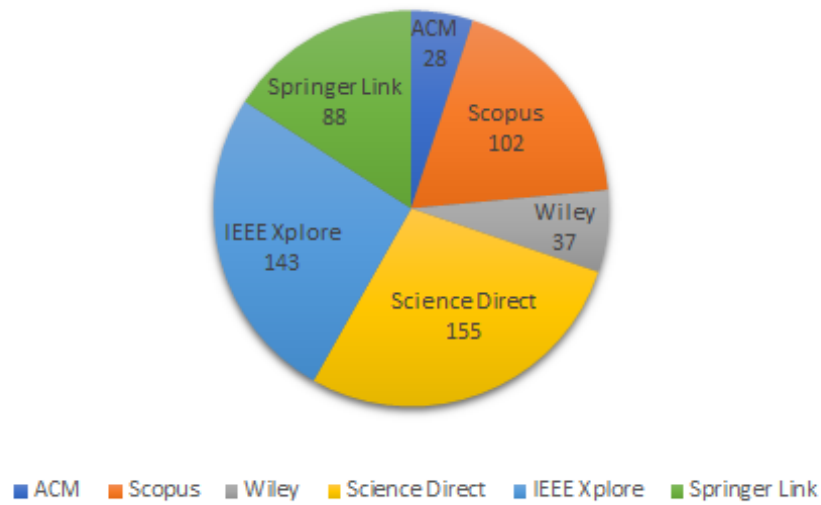
Durante a fase de execução os estudos são selecionados para a extração dos dados. O protocolo desta RSL foi definido em Setembro de 2020, bem como as fases de busca, leitura e seleção dos trabalhos ocorreram no mesmo período. As buscas na base IEEE Xplore obtiveram o total de 143 trabalhos, enquanto que 102 trabalhos foram retornados pelas buscas na base Scopus, 37 trabalhos na Wiley Online Library, 155 na Science Direct, 88 artigos encontrados na Springer Link e 28 estudos obtidos na base da ACM Digital Library. Os resultados das buscas são apresentados na Figura 11.

Os critérios de inclusão e exclusão definidos no protocolo de pesquisa foram aplicados primeiramente nos títulos dos trabalhos. Em seguida, foram analisados o *abstract*, introdução e conclusão dos estudos, resultando em um total de 15 documentos para serem lidos por completo e obtenção de dados. Além dos trabalhos selecionados encontrados nas bases de buscas, foi adicionado o trabalho (Salgueiro, 2009), o qual serviu como base para a elaboração desta dissertação.

## 3.3 Análise

Esta fase tem como objetivo sumarizar os dados, analisá-los e apresentá-los com a finalidade de demonstrar que os estudos selecionados atendem aos requisitos da RSL, ou seja, que respondam a pelo menos uma das questões de pesquisa. A Tabela 6 apresenta os trabalhos selecionados nesta etapa.

Figura 11 – Trabalhos encontrados por base de busca.



Fonte: (Autor, 2020)

Tabela 6 – Estudos Primários Seleccionados

ID	Ano	Título	Referência	Fonte
EP_01	2009	Alocação de Recursos com Justiça: Uma Aplicação de Jogos Cooperativos em Redes de Computadores	(Salgueiro, 2009)	UFPE
EP_02	2016	Software-defined dynamic bandwidth optimization (SD-DBO) algorithm for optical access and aggregation networks	(Zhao et al., 2016)	Springer
EP_03	2019	On SDN-Enabled Online and Dynamic Bandwidth Allocation for Stream Analytics	(Aljoby et al., 2019)	IEEE
EP_04	2019	Software-Defined Dynamic Bandwidth Management	(Chang et al., 2019)	IEEE
EP_05	2015	A heuristic algorithm for optimal discrete bandwidth allocation in SDN networks	(Tang and Liu, 2015)	IEEE
EP_06	2017	Adaptive QoS for data transfers using software-defined networking	(Boley, 2016)	IEEE

EP_07	2019	End-to-end dynamic bandwidth resource allocation based on QoS demand in SDN	(Thazin and Nwe, 2019)	IEEE
EP_08	2019	SDN- based Dynamic Bandwidth Allocation for Multiple Video-on-Demand Players over HTTP	(Thinh et al., 2019)	IEEE
EP_09	2015	A priority based virtual network bandwidth guarantee method in software defined network	(Fu et al., 2015)	IEEE
EP_10	2020	A SDN/OpenFlow Framework for Dynamic Resource Allocation based on Bandwidth Allocation Model	(Torres et al., 2020)	IEEE
EP_11	2018	QoE-Aware bandwidth broker for HTTP adaptive streaming flows in an SDN-Enabled HFC network	(Begen, 2018)	IEEE
EP_12	2015	Joint allocation and scheduling of network resource for multiple control applications in SDN	(Feng and Wang, 2014)	IEEE
EP_13	2018	Research on Bandwidth Control Technology Based on SDN	(Wang and Shi, 2018)	IEEE
EP_14	2016	SDN Based QoE Optimization for HTTP-Based Adaptive Video Streaming	(Zhu et al., 2015)	IEEE
EP_15	2016	SDN-based efficient bandwidth allocation for caching enabled cognitive radio networks	(Hu et al., 2016)	IEEE

		Quality of service (QoS)-guaranteed network resource allocation via software defined networking (SDN)	(Akella and Xiong, 2014)	IEEE
EP_16	2014			

Fonte: (Autor, 2020)

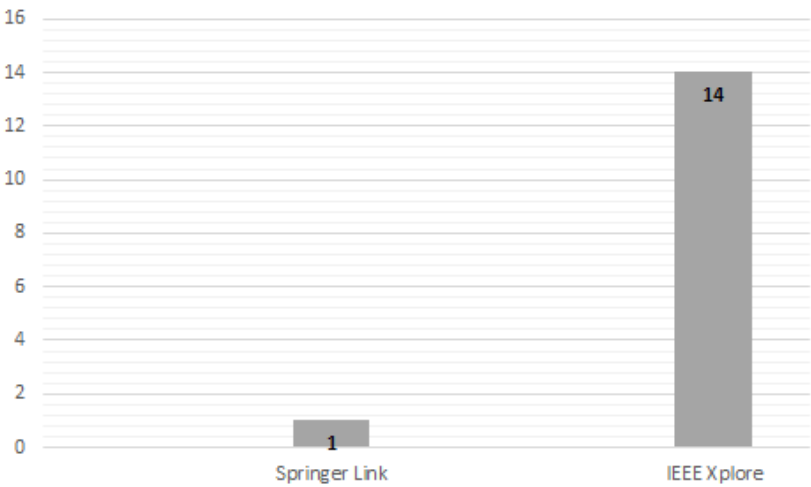
### 3.4 Resultados da Análise

Nesta etapa da revisão sistemática são apresentados os resultados quantitativos das bases de buscas, ano, tema de interesse, questões de pesquisa e citações das 15 publicações que foram aceitas de acordo com os critérios de inclusão e exclusão definidos nesta RSL.

#### 3.4.1 Base dos trabalhos

A Figura 12 apresenta os trabalhos selecionados categorizados por base de busca. Pode-se observar que a maior parte dos estudos foram obtidos da base IEEE Xplore, totalizando 14 trabalhos e representando 80% do total. Das bases Springer Link e Science Direct foram obtidos 2 artigos em cada, somando 16% do total de artigos. A base com menor representatividade foi a ACM Digital Library, onde apenas 1 trabalho foi obtido, representando 4% dos trabalhos selecionados.

Figura 12 – Quantidade de estudos selecionados por base de busca.



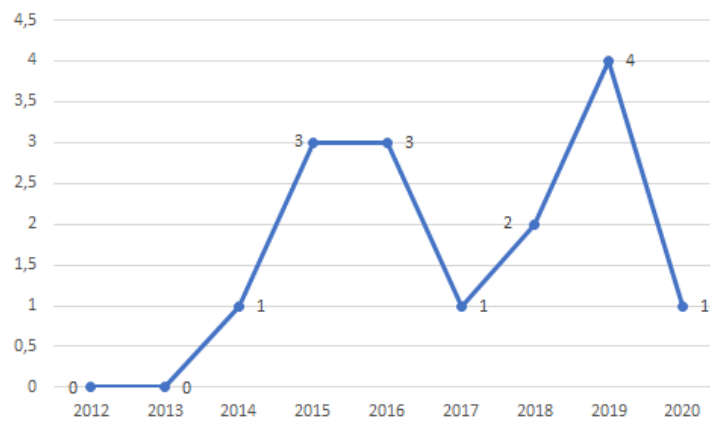
Fonte: (Autor, 2020)



### 3.4.2 Ano das Publicações

Os anos das publicações são ilustrados na Figura 13. Pode-se notar que o ano do trabalho mais antigo dentre os selecionados é datado de 2014 e que nos anos seguintes, 2015 e 2016 também houveram trabalhos classificados para esta RSL, com 3 publicações em cada ano. No ano de 2017 houve apenas 1 publicação, enquanto que em 2018 houve o acréscimo de uma publicação. A maioria dos trabalhos escolhidos foram publicados em 2019, totalizando 4 publicações e representando 27% dos estudos. No ano de 2020 houve novamente um decréscimo em relação ao ano anterior resultando em apenas 1 publicação. Não houveram publicações nos anos de 2012 e 2013. Diante destes resultados é possível concluir que ainda existem poucos estudos relativos ao foco desta RSL, constatando ainda que existem diversos campos a serem explorados.

Figura 13 – Quantidade de trabalhos selecionados por ano de publicação.

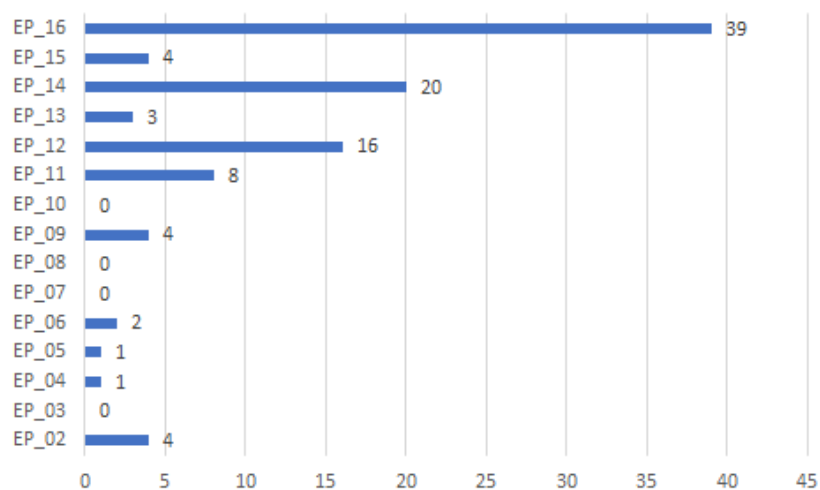


Fonte: (Autor, 2020)

### 3.4.3 Citações das Publicações

Também foi realizado um levantamento acerca das citações das publicações. A Figura 14 ilustra a quantidade de citações que cada publicação recebeu até o momento da realização desta RSL. Destaca-se o trabalho EP\_16 (Akella and Xiong, 2014) por ter obtido o maior número de citações, 39 no total, seguido por EP\_14 (Zhu et al., 2015) e EP\_12 (Feng and Wang, 2014) com 20 e 16 citações, respectivamente. O trabalho EP\_11 (Begen, 2018) também merece destaque por obter 8 citações. Os estudos EP\_02 (Zhao et al., 2016), EP\_09 (Fu et al., 2015) e EP\_15 (Hu et al., 2016) receberam 4 citações cada um, seguidos pelos trabalhos EP\_13 (Wang and Shi, 2018), EP\_06 (Boley, 2016), EP\_05 (Tang and Liu, 2015) e EP\_04 (Chang et al., 2019) recebendo 3, 2, 1 e 1 citações, respectivamente. Os estudos EP\_03 (Aljoby et al., 2019), EP\_07 (Thazin and Nwe, 2019), EP\_08 (Thinh et al., 2019) e EP\_10 (Torres et al., 2020), provavelmente, por terem sido publicados mais recentemente, ainda não possuem nenhuma citação.

Figura 14 – Quantidade de citações por publicação.



Fonte: (Autor, 2020)

## 3.5 Respostas às questões de pesquisa

Esta RSL tem como objetivo conhecer o estado da arte em modelos de alocação de banda em controladores SDN. A Tabela 7 apresenta os trabalhos e quais das questões de pesquisa eles respondem. Todos os trabalhos respondem pelo menos a uma das questões de pesquisa definidas, porém nem todos que foram classificados responderam a todas elas.

Tabela 7 – Questões de pesquisa respondidas por trabalho.

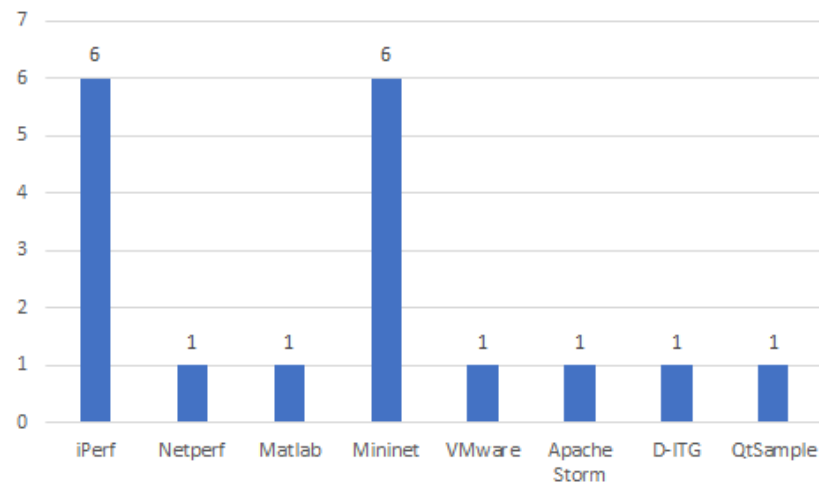
ID	QP1	QP2	QP3
EP_01			x
EP_02	x	x	
EP_03	x	x	x
EP_04	x	x	x
EP_05	x	x	
EP_06	x	x	x
EP_07	x	x	x
EP_08	x	x	x
EP_09	x	x	
EP_10	x	x	
EP_11	x	x	x
EP_12	x	x	
EP_13	x	x	x
EP_14	x	x	x
EP_15	x	x	
EP_16	x	x	

Fonte: (Autor, 2020)

### 3.5.1 QP1

Dentre os trabalhos selecionados, todos respondem a primeira questão de pesquisa (**QP1**), ou seja, utilizaram ferramentas para validar seus experimentos. Pode-se observar na Figura 15 que a ferramenta *iPerf*, software utilizada para geração de fluxo de datagramas, foi a ferramenta mais utilizada na fase de experimentação dos artigos. Outra ferramenta utilizada com maior frequência foi o *Mininet*, *software* para simulação de topologia de rede. Na grande maioria dos trabalhos que utilizaram o *Mininet* como simulador, também foi feito uso do *iPerf*.

Figura 15 – Resposta QP1.

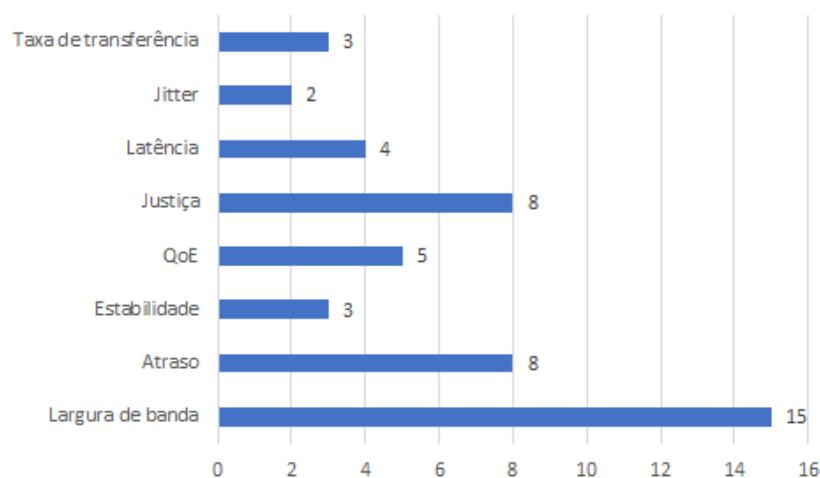


Fonte: (Autor, 2020)

### 3.5.2 QP2

Esta questão de pesquisa foi respondida por todos os trabalhos analisados, ou seja, em todos os artigos os autores utilizaram alguma métrica para a análise, validação da qualidade e eficiência de seus trabalhos. Pode-se observar na Figura 16 quais métricas foram analisadas pelos autores, onde a grande maioria tomou como base para análise a largura de banda utilizada, presente em 15 trabalhos. Em seguida, as métricas mais utilizadas foram atraso, justiça, latência e qualidade de experiência do usuário (QoE). Outras métricas utilizadas pelos autores foram o jitter, estabilidade e taxa de transferência.

Figura 16 – Resposta QP2.

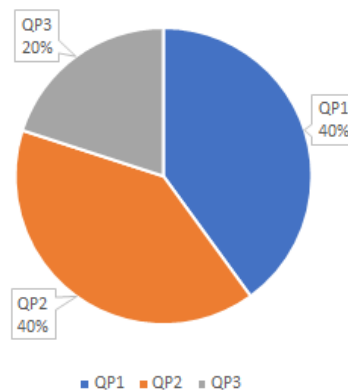


Fonte: (Autor, 2020)

### 3.5.3 QP3

Entre todos os trabalhos selecionados, apenas 8 responderam a esta questão de pesquisa, ou seja, implementam alguma estratégia para alocação de largura de banda em um controlador SDN, o que reforça ainda mais a justificativa deste trabalho de dissertação. Os estudos que responderam esta questão de pesquisa utilizam técnicas de criação de políticas de QoS para melhorar ou otimizar recursos, porém em nenhum dos casos é feita a implementação de mecanismos que possibilitem a criação de políticas de QoS e compartilhamento de banda diretamente por um controlador SDN. Por fim, a Figura 17 representa as questões de pesquisa e a porcentagem dos trabalhos que as respondem.

Figura 17 – Questões de pesquisa e a porcentagem de trabalhos que as respondem



Fonte: (Autor, 2020)

## 3.6 Considerações Finais

Este trabalho realizou uma revisão sistemática da literatura sobre modelos de alocação de banda em Redes Definidas por Software. Foram coletados e analisados dados de 15 trabalhos resultantes. Os resultados obtidos por essa revisão sistemática podem auxiliar pesquisadores em trabalhos futuros relacionados ao tema. Para administradores de redes, as técnicas e ferramentas utilizadas nos trabalhos podem servir de base para implementação de soluções que objetivem melhorar o desempenho da infraestrutura.

No próximo Capítulo são apresentados os principais trabalhos relacionados esta pesquisa resultantes desta revisão sistemática.

# 4

## Trabalhos Relacionados

Este capítulo apresenta os trabalhos relacionados à esta dissertação. Os estudos aqui apresentados são resultantes da revisão sistemática. Todos os trabalhos analisados na revisão sistemática, de alguma forma possuem relação com o tema deste trabalho, seja pelo uso de políticas de QoS para otimização de recursos de rede ou por apresentar arquiteturas em SDN com a proposta de melhoria de desempenho.

Alguns trabalhos apresentaram abordagens semelhantes e por esta razão foram agrupados. Esses trabalhos foram separados em grupos que abordam temas com maior relação a este trabalho: estudos que abordam modelos de alocação de banda e trabalhos relacionados à redes definidas por software. A próxima seção apresenta estes trabalhos.

### 4.1 Modelos de alocação de banda

[Salgueiro \(2009\)](#) apresenta uma metodologia para alocação de largura de banda de forma justa em fluxos de tráfego distintos em redes multisserviço, aplicando conceitos da *Teoria dos Jogos Cooperativos*. Neste trabalho são propostos também dois novos modelos de alocação de recursos: MARS - MAR with Shapley value e SHARM - SHApley value with Reservation Model. Estes novos modelos utilizam o modelo MAR como regra de divisão, mas também utilizam outros modelos matemáticos como base, no caso, utilizam modelos econômicos como regra de alocação em vez de usar a regra de divisão proporcional.

Esses modelos podem ser implementados em redes SDN, desde que sejam implementados também mecanismos de diferenciação de serviços.

As Seções [2.3.4](#) e [2.3.5](#) do Capítulo 2 desta dissertação apresentam, respectivamente, os conceitos dos modelos MARS e SHARM de forma mais detalhada.

## 4.2 Algoritmos heurísticos

[Zhao et al. \(2016\)](#) propõe um algoritmo responsável por otimizar o uso da largura de banda de forma dinâmica considerando as alterações de tráfego em tempo real em redes de agregação sob uma arquitetura SDN. O algoritmo proposto atua coletando e analisando informações em tempo real e gera uma estratégia de otimização da largura de banda para toda a rede. São utilizadas também estratégias de previsão para reduzir os efeitos dos atrasos no processamento através de cálculos utilizando o modelo NeuroFuzzy ([Zhani et al., 2009](#)). O algoritmo foi analisado em simulação para validar o desempenho em termos de taxa de utilização de recursos, média de atraso e comparar também a utilização e não utilização das estratégias de previsão. Os resultados apresentados indicam um melhor desempenho na alocação de largura de banda e maior eficácia na utilização da rede.

[Aljoby et al. \(2019\)](#) apresenta um algoritmo heurístico para prover uma solução de alocação de recursos próxima do ótimo. Este algoritmo foi implementado em um módulo do controlador SDN *OpenDaylight* ([OpenDaylight, 2020](#)) tendo como base para análise o Apache Storm, que é uma plataforma de código aberto para processamento de fluxo em tempo real e ilimitado. O algoritmo trabalha capturando alterações no sistema e coletando a medição dos fluxos em tempo real. A aplicação dos comandos de otimização da largura de banda se dão através de um componente implementado na solução que coleta as estatísticas do link e envia para o controlador SDN, que por sua vez, possui uma implementação responsável por receber as informações dos links, calcular as taxas de alocação de recursos, converter em mensagens OpenFlow e instalar nos dispositivos da rede.

Para resolver o problema de alocação de largura de banda de forma justa em redes multicast sempre que uma nova sessão de usuário é iniciada, o trabalho apresentado por ([Tang and Liu, 2015](#)) propõe um algoritmo heurístico com a finalidade de evitar que todas as alocações da rede sejam recalculadas neste momento, adiando a realização dos ajustes apenas quando não houver mais largura de banda livre disponível o suficiente, reduzindo assim a sobrecarga e no momento mais oportuno realizar uma alocação de recursos justa. O algoritmo foi implementado em uma estrutura SDN atuando de forma que, se for necessário, os reajustes de alocação de recursos serão aplicados através de entradas na tabela de fluxos do Open vSwitch.

## 4.3 Políticas de QoS

Em ([Boley, 2016](#)) foi desenvolvido um algoritmo para controlar em tempo real a alocação de largura de banda baseando-se no fluxo e na prioridade de classes de tráfego usando os princípios dos serviços diferenciados clássicos (*DiffServ*). O algoritmo foi implementado como uma extensão de um controlador SDN dividido em componentes que possuem responsabilidades distintas, tais como as atividades de monitoramento, análise e controle. Um destes componentes é o responsável por analisar e identificar os fluxos que estão sendo executados com capacidade

abaixo das suas taxas reservadas. Ao detectar largura de banda não utilizada, a parte ociosa é adicionada a um *pool* de largura de banda para ser utilizada por outros fluxos ao longo da rede. Outro componente é responsável por identificar os fluxos que estão utilizando taxas abaixo de suas reservas, mas estão aumentando. No caso de haver largura de banda insuficiente, o algoritmo obtém largura de banda disponível dos fluxos de prioridade mais baixa de acordo com as políticas de QoS.

Thazin and Nwe (2019) propõe um esquema de alocação de largura de banda dinâmica baseado em QoS em SDN de acordo com a prioridade dos fluxos. Uma das principais características do esquema proposto é o uso do mecanismo de enfileiramento do protocolo OpenFlow. O esquema foi implementado em dois níveis: no nível do controlador e no nível dos switches. A nível do controlador, o cliente envia uma solicitação ao controlador incluindo informações de origem e destino, largura de banda e um valor corresponde a tolerância de atraso. Quando o controlador recebe a solicitação, calcula a rota de acordo com a largura de banda da solicitação utilizando o mecanismo de monitoramento da topologia. Após calcular a rota mais viável, o controlador mapeia o fluxo em filas pré criadas nas portas dos switches e cria as regras de encaminhamento em cada switch da rota determinada. Os experimentos foram executados em uma topologia virtual criada no software Mininet e conectada no controlador SDN Ryu (RYU, 2020).

## 4.4 Estratégias de Alocação de Banda em SDN

Begen (2018) apresenta uma solução para alocação de largura de banda em SDN com reconhecimento de QoE (*Quality of Experience*). A solução é composta por vários componentes localizados no plano de dados, no plano de controle e no plano de aplicação. O componente do plano de aplicação possui agentes atuando na coleta de estatísticas de rede, na estimativa da largura de banda utilizando um algoritmo denominado PANDA (Li et al., 2014) e um agente responsável por tomadas de decisões baseadas em cálculos para otimizar o QoE do cliente. Existe também um agente responsável pela comunicação entre o plano de aplicação e o plano de controle, que é realizada através da API REST do controlador de rede. Foi implementado também no controlador um serviço incumbido de receber as informações e aplicar a alocação dinâmica de largura de banda. As mensagens recebidas no controlador são traduzidas em mensagens OpenFlow e são instaladas nos dispositivos de encaminhamento de fluxo da rede.

Wang and Shi (2018) adiciona uma estratégia de controle de largura de banda em um controlador SDN baseando-se na prioridade dos usuários utilizando a tabela de fluxos do Open vSwitch. Tal estratégia consiste na implementação de um módulo no controlador Floodlight para realizar o monitoramento de fluxo e controle da largura de banda. Faz parte também desta estratégia a criação de filas no Open vSwitch via linha de comando. O mecanismo de controle de alocação de recursos dá-se através de chamadas de funções nativas do Open vSwitch pelo controlador.



[Chang et al. \(2019\)](#) apresenta uma técnica de alocação de recursos onde o controlador SDN aplica de forma dinâmica a alocação de largura de banda previamente agendada. Esta abordagem proposta faz também o uso da arquitetura OpenWrt ([Dutt et al., 2012](#)). Um modulo de controle foi desenvolvido para o controlador OpenFlow responsável por coletar e monitorar informações dos fluxos e os status de cada nó da rede. As informações coletadas são usadas para verificar a largura de banda usada por cada nó, em seguida, os dados analisados são enviados para um mecanismo de tomada de decisão baseado em políticas de alocação de largura de banda para alterar as regras de controle na tabela de fluxo de todos os nós da rede através de um canal seguro.

## 4.5 Cosiderações Finais

Este capítulo apresentou os trabalhos relacionados ao gerenciamento de largura de banda em redes definidas por software. Os vários estudos relacionados mostram a relevância deste trabalho. Em um primeiro momento, foram analisados alguns trabalhos que abordam mecanismos de gerenciamento de largura de banda e arquiteturas propostas para otimização de recursos em SDN. A necessidade de uma melhor compreensão destes mecanismos justifica a escolha destes trabalhos para uma análise mais aprofundada. No entanto, entre os trabalhos analisados, em nenhum são propostos mecanismos que utilizem modelos de alocação de banda com critérios de justiça implementados em um controlador SDN para o compartilhamento de recursos.

A Tabela 8 apresenta as questões relacionadas ao tema deste trabalho. Os artigos apresentados neste capítulo, serviram como arcabouço intelectual e indicação para possíveis rumos para esta pesquisa. É válido salientar que, as buscas foram realizadas nos repositórios mencionadas na Revisão Sistemática deste trabalho, filtrados pelo período de 2012 a 2020. Não foram encontrados trabalhos que abrangem todos os temas abordados por esta dissertação mostrados na Tabela 8.

Tabela 8 – Trabalhos relacionados

Artigo	SDN	QoS	Compartilha largura de banda com justiça	Implementa BCM	Cria Modulo Controlador
Salgueiro (2009)	X	X	X		
Zhao et al. (2016)	X				
Aljoby et al. (2019)	X				X
Chang et al. (2019)	X				
Tang and Liu (2015)	X		X		X
Boley (2016)	X	X			X
Thazin and Nwe (2019)	X	X			
Begen (2018)	X				X
Wang and Shi (2018)	X				X
Oliveira (2020)	X	X	X	X	X

Fonte: (Autor, 2020)

# 5

## Modelo de Alocação de Banda com Critério de Justiça

Neste Capítulo é apresentado um modelo de alocação de recursos baseado em critérios de justiça com a finalidade de oferecer uma divisão justa de largura de banda entre classes de tráfego distintas.

### 5.1 Conceitos de justiça em Redes de Computadores

As redes de computadores são projetadas partindo da ideia de que seu desempenho seja otimizado e que seus custos sejam minimizados através do compartilhamento de recursos, de modo que, todos os usuários ou aplicações tenham direito aos recursos, desde que o acesso simultâneo não prejudique os demais usuários ([Salgueiro, 2009](#)).

Nos trabalhos que tratavam de justiça durante a década de 80, as noções predominantes àquela época eram de que nenhum usuário deveria ser proibido de acessar a rede, um usuário poderia usar os recursos de rede tanto quanto qualquer outro usuário, em caso de congestionamento, o usuário que estivesse consumindo mais recursos teria o acesso mais restrito em relação aos outros usuários que não estivessem gerando sobrecarga na rede.

Com o aumento da complexidade dos serviços em rede, surgiu a necessidade de implementação de mecanismos de diferenciação de serviços capazes de detectar as características, requisitos de desempenho e nível de congestionamento de cada fluxo. A necessidade de implantação de novos serviços forçaram os controles de QoS a adotarem novos conceitos de equidade, incluindo nos mecanismos de controle o tratamento de proporcionalidade e prioridade.

Os três tipos mais comuns de implementação de mecanismos com equidade em termos de redes de computadores são:

- **Mecanismos de controle de fluxo fim-a-fim:** projetados para que os fluxos utilizem toda a capacidade da rede de forma justa. O mecanismo de equidade mais difundido em redes

de computadores é o mecanismo Max-min. Este mecanismo garante a alocação de fluxo mínima para cada fluxo levando em consideração suas diferenças.

- **Mecanismos de escalonamento:** estes mecanismos implementam políticas associadas a sistemas de filas. Os dados são armazenados em *buffers* e quando os canais estão prontos para realizar a transmissão desses dados, uma política de escalonamento baseada em algum critério de justiça determina qual pacote deve ser transmitido.
- **Gerenciamento de recursos nos nós da rede:** para garantir a diferenciação dos serviços, os administradores de rede devem especificar para cada nó como os recursos devem ser compartilhados. Modelos de alocação de banda determinam como os recursos são compartilhados entre as classes com demandas diferentes, de acordo com os critérios de justiça definidos por quem projetou a rede.

Modelos de alocação de banda definem a forma de como os recursos são distribuídos em situações de conflito de compartilhamento. A adoção do modelo e o critério de distribuição de recursos depende das classes de tráfego presentes na rede e de seus perfis.

As classes de tráfego podem ser caracterizadas de forma individualizada, porém a definição de justiça de alocação de recursos continua em aberto ([Salgueiro, 2009](#)).

Neste trabalho aplicam-se critérios de justiça definidos pelo autor, para a criação de regras com a finalidade de resolver os problemas de alocação e prover uma distribuição de recursos justa em termos de largura de banda em redes de computadores. Como resultado da aplicação destes critérios, foi desenvolvido um modelo de alocação de banda, o MAMDFS - *Maximum Allocation Model With Dynamic Fairness Sharing*, visando a diminuição de desperdício e a otimização dos recursos através do compartilhamento de banda entre CTs. Na próxima seção são apresentadas as regras de alocação e compartilhamento deste modelo.

## 5.2 MAMDFS - Maximum Allocation Model With Dynamic Fairness Sharing

Nesta Seção é apresentado um novo modelo de alocação de recursos: MAM com compartilhamento dinâmico justo ( MAMDFS - *Maximum Allocation Model with Dynamic Fairness Sharing*). Seu desenvolvimento foi baseado no modelo clássico MAM, apresentado na Seção 2.3.1. Suas regras de alocação são definidas nas próximas seções.

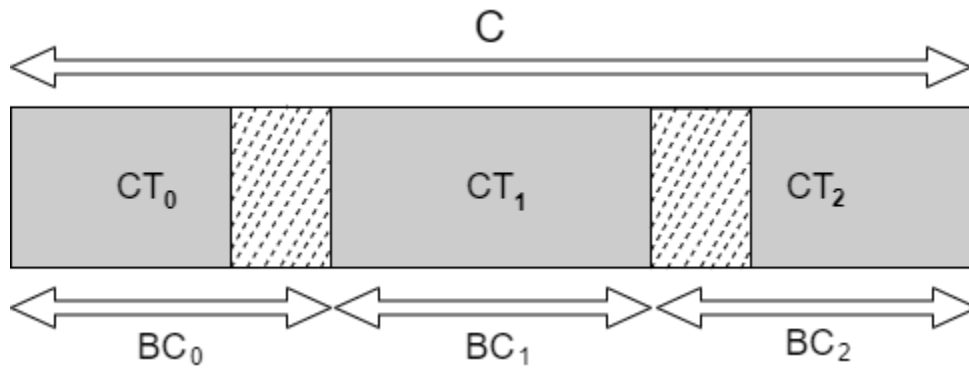
## 5.3 Regras de alocação e compartilhamento

As regras de alocação definidas para o MAMDFS distinguem-se do MAM em dois pontos principais:

1. O somatório de todas as BCs não pode exceder o tamanho máximo da largura de banda disponível;
2. No MAMDFS além de ser permitido o compartilhamento, uma CT pode utilizar largura de banda deixada sem uso por outra CT.

Para auxiliar no entendimento, as figuras Figura 18 e Figura 19 exemplificam a representação canônica do MAMDFS. Onde  $C$  é a capacidade máxima que pode ser alocada para um *link*,  $BC_i$  representa a capacidade máxima que pode ser alocada para  $CT_i$ , enquanto que a área tracejada representa a largura de banda livre de  $CT_i$ .

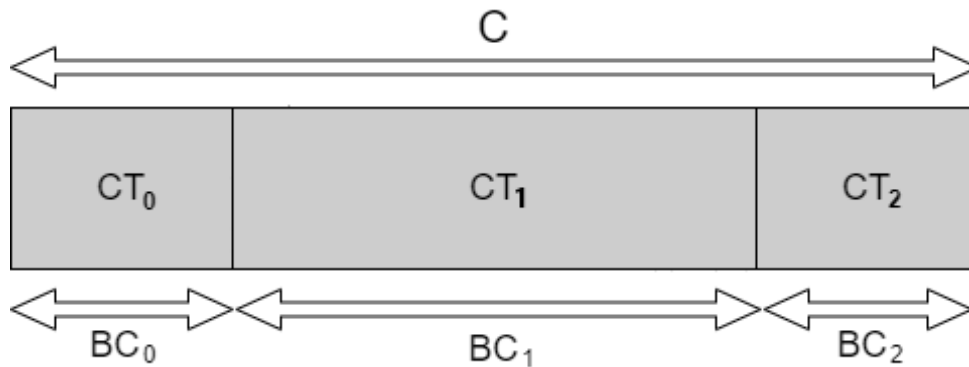
Figura 18 – Representação canônica MAMDFS (Autor)



Fonte: (Autor, 2020)

Pode-se observar na Figura 19 que as BCs foram realocadas, fazendo com que  $CT_1$  obtivesse a largura de banda não utilizada pelas outras CTs. Observa-se também que a quantidade de largura de banda alocada por todas as BCs não excede  $C$ .

Figura 19 – Representação canônica MAMDFS BCs realocadas (Autor)



Fonte: (Autor, 2020)

O MAMDFS também impõe limites para a alocação e o compartilhamento de recursos entre as CTs.

As regras de alocação e compartilhamento estabelecem que:

- Uma CT não deve exceder o tamanho de sua BC;
- A soma de todas as bandas reservadas para todas as CTs não deve ultrapassar o tamanho total do *link*;
- O empréstimo de banda entre as CTs somente é permitido caso haja banda livre disponível;
- A CT com saturação de banda, isto é, com perda de pacotes, primeiramente tenta obter de volta a banda anteriormente emprestada para outras CTs. Caso nenhuma outra CT esteja em situação de dívida, ou seja, não tenha tomado por empréstimo largura de banda, então esta obtém por empréstimo largura de banda de outra CT com a maior quantidade de banda livre disponível entre todas as outras;
- O empréstimo de banda deve ocorrer a partir do momento em que haja perda de pacotes durante a transmissão de fluxo nas CTs;

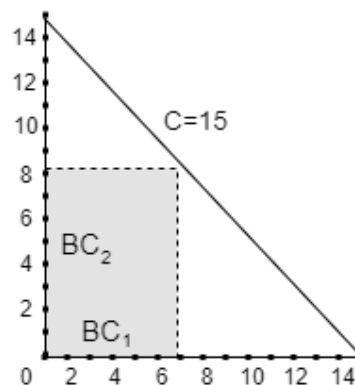
O MAMDFS pode ter sua representação canônica formulada pela seguinte equação:

$$\begin{cases} \sum_0^i X_i \leq C \\ X_i \leq BC_i \end{cases}$$

onde,  $X_i$  representa a banda reservada para todos os LSPs que pertencem a um mesmo  $CT_i$ .

Outro exemplo de alocação para duas CTs pode ser observado na Figura 20, onde  $BC_1 = 7$  e  $BC_2 = 8$ . Observa-se também que a soma dos BCs não ultrapassam  $C$ .

Figura 20 – Regra de Alocação MAMDFS (Autor)



Fonte: (Autor, 2020)

Diante da definição das regras de alocação e compartilhamento formuladas para este modelo, pode-se observar que a aplicação do conceito de justiça se dá pelo fato de que, não haverá banda ociosa ou livre em uma ou mais CTs, quando uma ou mais CTs alcançarem os limites de banda estabelecidos através do compartilhamento de banda livre, evitando assim desperdícios e realizando uma distribuição justa dos recursos disponíveis.

## 5.4 A arquitetura *BW-Manager*

Após a definição do modelo de alocação de banda com justiça, foi desenvolvida uma arquitetura denominada *BW-Manager* (*Bandwidth Manager*) composta por vários componentes em um módulo para o controlador SDN *Floodlight* ([Floodlight, 2020](#)), capaz de realizar o monitoramento de CTs e aplicar as regras de restrição e compartilhamento definidas no MAMDFS.

A escolha do *Floodlight* para a realização deste trabalho se deu por conta de sua grande adoção pelas comunidades acadêmica e de desenvolvedores. Além disso, seu projeto é de código aberto, conta com uma vasta documentação e possui uma comunidade bastante ativa. Outro fator que pesa positivamente em seu favor é o baixo consumo de recursos de *hardware* e fácil configuração.

Como mostrado na Figura 4, um controlador SDN tem a capacidade de estender suas funcionalidades através da implementação de módulos adicionais, que podem ser desenvolvidos diretamente no controlador a partir do seu código fonte ou através de aplicações que se comunicam com a camada de controle através de uma REST API.

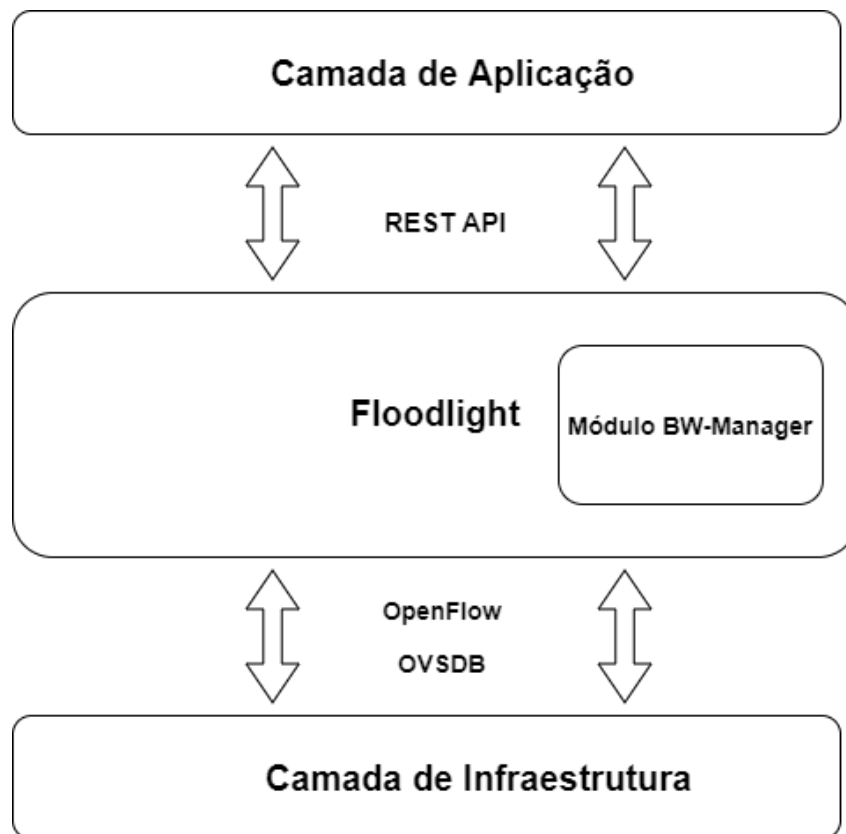
Neste trabalho, foi adicionado a um controlador SDN um módulo com a capacidade de aplicar políticas de QoS e realizar o compartilhamento de recursos através da implementação de modelos de alocação de banda. A Figura 21 apresenta uma visão macro da arquitetura envolvendo as camadas de um ambiente SDN com o novo módulo implantado na camada de controle.

A arquitetura *BW-Manager* é composta por um projeto de *software*, que por sua vez, é composto por um conjunto de componentes menores que desempenham tarefas distintas. Esta implementação é possível graças às interfaces programáticas fornecidas pelo controlador especificamente para esta finalidade.

Os componentes da arquitetura e suas respectivas responsabilidades, compreendem-se como:

- a) **Endpoint Parâmetros:** É o componente responsável por receber todos os parâmetros necessários para iniciar o mecanismo de monitoramento. Após recebimento dos parâmetros, realiza também a validação das informações antes de acionar o núcleo do módulo. Este serviço fica disponível desde a inicialização do controlador.

Figura 21 – Visão macro da arquitetura (Autor).



Fonte: (Autor, 2020)

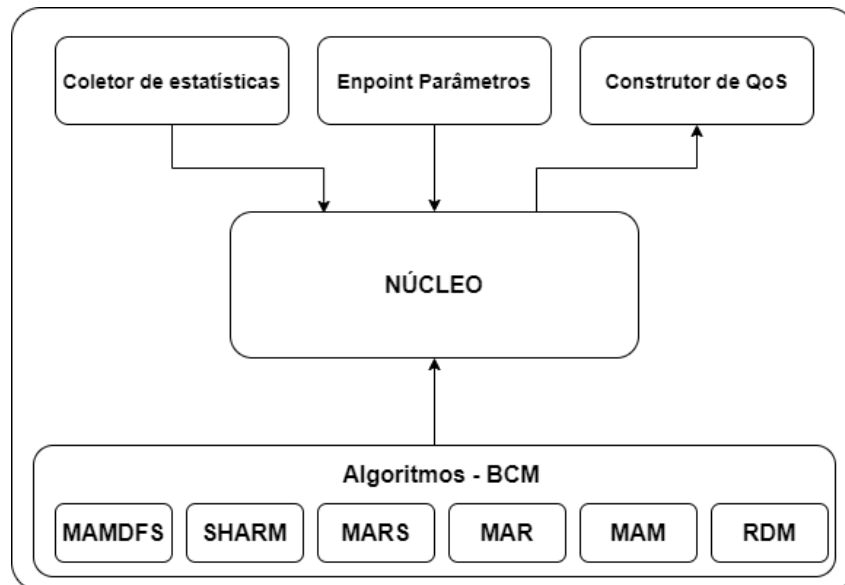
- b) Coletor de estatísticas:** Trata-se do componente responsável por monitorar e coletar as estatísticas de fluxo. A coleta de estatísticas acontece em um intervalo de tempo fornecido por parâmetro.
- c) Construtor de QoS:** Este componente tem a responsabilidade de construir os comandos com base nos parâmetros fornecidos pelo Núcleo e aplicar as políticas de QoS. É o responsável também por aplicar os comandos necessários para realizar as atividades de compartilhamento de largura de banda. Os comandos são aplicados através do protocolo OVSDB.
- d) Algoritmos - BCM:** É a biblioteca onde são implementados os algoritmos usados pelo Núcleo para aplicar as políticas de QoS e compartilhamento de recursos com base nos modelos de alocação de banda.
- e) Núcleo:** Este é o componente mais importante do módulo. É o componente que encapsula a inteligência da arquitetura. Seu funcionamento se baseia nas atividades dos outros componentes para tomar as decisões. Após obter os parâmetros de inicialização, aciona o Construtor de QoS para aplicar as políticas fornecidas pelo *Endpoint* de Parâmetros. Após aplicar as políticas de QoS, aciona o mecanismo de monitoramento e coleta de estatísticas.



Em seguida, com base nas estatísticas e os parâmetros recebidos, toma a decisão de realizar, ou não, o compartilhamento de largura de banda entre as CTs.

A Figura 22 apresenta a arquitetura *BW-Manager* e seus componentes.

Figura 22 – Componentes de arquitetura *BW-Manager* (Autor).



Fonte: (Autor, 2020)

## 5.5 Implementação do modelo MAMDFS

A fim de validar o modelo MAMDFS, foi implementado um algoritmo seguindo as suas regras de alocação e compartilhamento.

O algoritmo recebe como entrada informações sobre as filas que serão monitoradas. Estas informações contém o ID, tamanhos mínimo e máximo de largura de banda disponível para cada uma das filas.

O processo de monitoramento é iniciado assim que os dados de entrada são enviados ao controlador. Os dados são recebidos pelo controlador através de sua API REST em um *endpoint* disponibilizado pelo módulo criado como parte deste trabalho. Ao receber as informações acerca das filas, o módulo cria as filas nas portas dos *switchs* e aplica as políticas de QoS através do protocolo OVSDb.

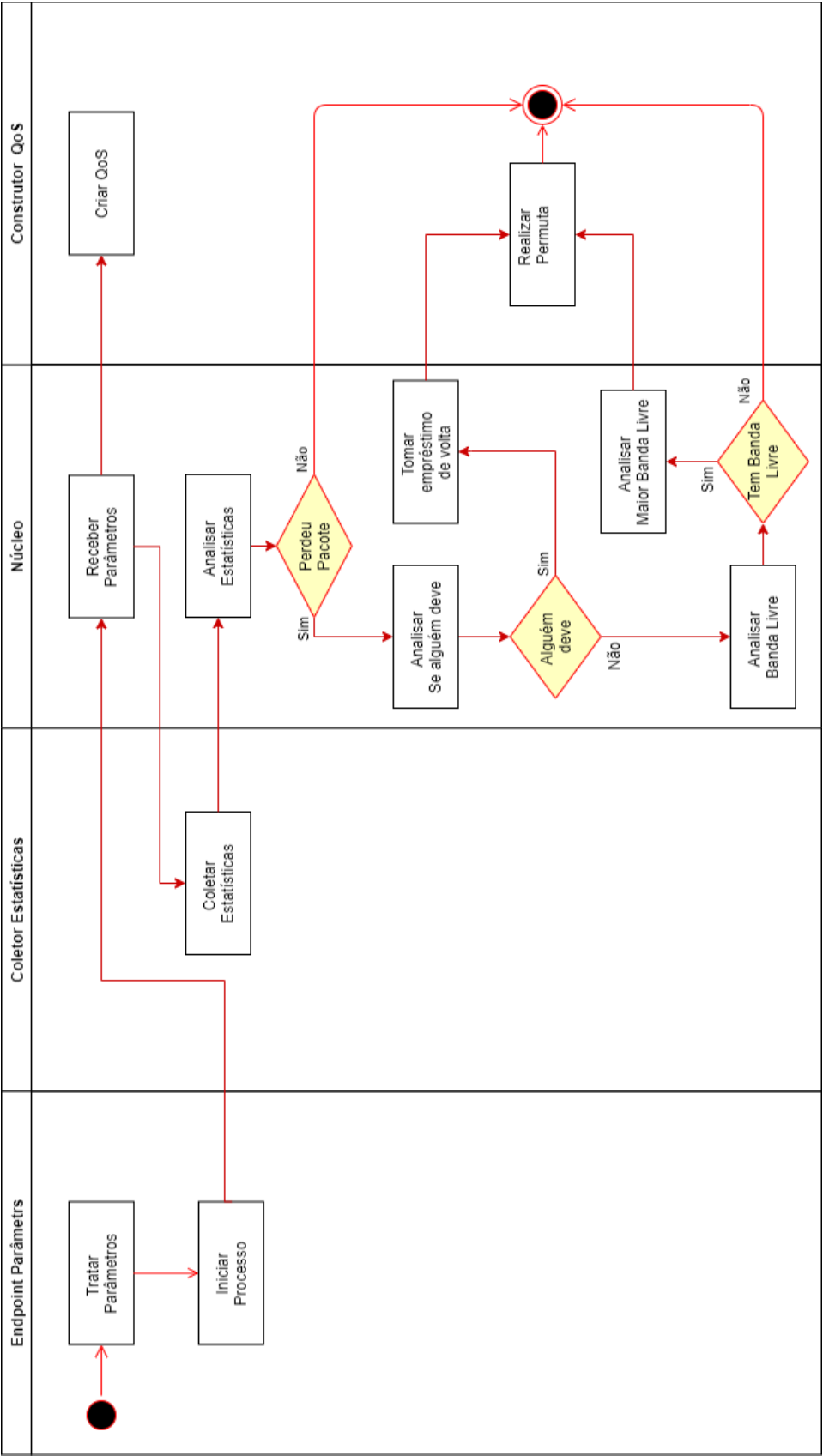
Além das informações sobre as filas e políticas de QoS, o algoritmo precisa também de parâmetros contendo informações sobre o intervalo de tempo em que serão coletadas as estatísticas dos fluxos nas filas. Outro parâmetro necessário para o funcionamento do mecanismo de permuta de largura de banda entre as filas é o valor referente à porcentagem que será tomado

de empréstimo das filas elegíveis para emprestar largura de banda livre. A seguir, tem-se um passo a passo que descreve detalhadamente o funcionamento do algoritmo.

- O servidor (controlador) recebe as informações sobre as filas, políticas de QoS, os parâmetros de tempo para coleta de estatísticas e porcentagem de empréstimo de largura de banda através da REST API;
- O Criador de QoS cria as filas nas portas dos switches e aplica as políticas de QoS através do protocolo OVSDB;
- O algoritmo inicia o monitoramento dos fluxos nas filas;
- No intervalo de tempo determinado por parâmetro obtém as estatísticas de fluxo das filas;
- Se houver perda de pacotes inicia o processo de barganha ([Carvalho, 2017](#));
- O algoritmo verifica se alguma outra fila está em situação de dívida com a fila que está perdendo pacotes e, em caso positivo, toma de volta a banda emprestada. Caso nenhuma outra fila esteja em situação de dívida, o algoritmo verifica se há largura de banda livre disponível entre as outras filas;
- Se houver largura de banda livre verifica em qual fila existe maior disponibilidade de largura de banda;
- Obtém por empréstimo da fila com maior banda livre uma porcentagem de acordo com o valor informado por parâmetro;
- Incrementa o tamanho da fila com perda de pacotes e subtrai o mesmo tamanho da fila que cedeu o empréstimo, respeitando assim o tamanho máximo do link;
- Caso não haja banda livre disponível para empréstimo ocorrerá perda de pacotes até surgir banda livre disponível;
- O processo se repete durante o funcionamento do controlador;
- Caso o valor do parâmetro referente a porcentagem de empréstimo seja zero, o processo de empréstimo de largura de banda não é acionado.

Para auxiliar no entendimento sobre o funcionamento do módulo, a Figura 23 apresenta o fluxo das atividades realizadas durante sua execução.

Figura 23 – Diagrama de atividades do módulo BW-Manager.



Fonte: (Autor, 2020)

O Algoritmo 1 representa o funcionamento do mecanismo de monitoramento das filas e empréstimo de largura de banda.

Algoritmo 1 – Monitoramento e aplicação de permuta de largura de banda

**Data:** porcentagemPermuta, tempoObterEstatisticasFilas, infoFilas[]

**Result:** Void

```

1 início;
2 while controlador em execução do
3   leia estatisticasFilas(infoFilas[]);
4   if existePerdaDados then
5     if seAlguemDeve() then
6       obtemValorEmprestado();
7     else
8       consultaFilaMaiorBandaLivre();
9       if existeFilaComBandaLivre then
10        tamanhoFilaComPerda <- (tamanhoFilaComPerda +
11          porcentagemPermuta);
12        tamanhoFilaComMaiorBandaLivre <-
13          (tamanhoFilaComMaiorBandaLivre - porcentagemPermuta);
14        end
15      end
16    end
17  end
18 end

```

Fonte: (Autor, 2020)

No Capítulo seguinte é apresentado um projeto de experimentação utilizando o modelo MAMDFS a fim de comprovar a viabilidade de sua aplicação, bem como sua eficiência.

# 6

## Experimentação

Neste Capítulo é apresentado um projeto de experimento desenvolvido com o intuito de validar o modelo de alocação de recursos com justiça e o módulo para o controlador SDN construídos neste trabalho.

Nas próximas Seções é apresentada a metodologia, preparação do ambiente, ferramentas utilizadas, execução e análise estatística dos dados.

### 6.1 Metodologia Experimental

A metodologia deste trabalho consistiu na realização de uma pesquisa explicativa por meio de uma revisão sistemática, como apresentada no Capítulo 3 e na execução de um experimento controlado, buscando avaliar o desempenho do modelo de controle de banda com justiça proposto. Baseado no empréstimo de um percentual de largura de banda ociosa de uma fila do link, esse modelo visa melhorar o aproveitamento total da largura de banda do link para oferecer Qualidade de Serviço (QoS) para os fluxos que o interpassam. O experimento foi organizado seguindo a abordagem *Goal Question Metric* (GQM), proposta por (Basili, Victor R., 1994), e o procedimento experimental de (Lima et al., 2018).

Antes da experimentação, foi preciso preparar o ambiente. O cenário utilizado foi um estudo de caso em SDN. Como o ponto principal do trabalho foi atestar o modelo proposto, a rede foi virtualizada utilizando o software emulador de rede *Mininet*, por fins de flexibilidade, agilidade e facilidade de experimentação. Como controlador SDN, foi escolhido o *Floodlight*, o qual foi recodificado para acoplamento de um módulo implementando a solução de compartilhamento de banda por empréstimo.

O experimento realizado objetivou analisar o aproveitamento da largura de banda de um link sem e com a utilização de um método de compartilhamento de banda (objeto de controle e de tratamento, respectivamente). O tratamento utilizado foi o método de compartilhamento

de banda por empréstimo, como proposto neste trabalho. Para verificar o desempenho do fator, foram injetados tráfegos de pacotes TCP e UDP na rede pela ferramenta *iPerf*, a fim de estressar o link alvo para coleta das métricas experimentais.

Por fim, foram aplicados testes estatísticos para validação dos dados e verificação das hipóteses formuladas. Teste de *Kolmogorov-Smirnov* (KS) para verificar a normalidade dos dados e o teste de *Wilcoxon* para validação das hipóteses também foram aplicados.

## 6.2 Planejamento experimental

Nesta seção são apresentados os objetivos dos experimentos realizados, contexto, suas variáveis e formulação de hipóteses.

### 6.2.1 Objetivo do experimento

O objetivo do experimento foi formalmente definido, com base no método GQM ([Basili, Victor R., 1994](#)), como segue: analisar o aproveitamento da largura de banda de um link pelo modelo de orquestração de banda baseado em permuta; com a finalidade de avaliar contra a não utilização de um modelo; com relação à quantidade de pacotes enviados e descartados, jitter e latência dos pacotes; do ponto de vista de engenheiros, arquitetos e administradores de rede; no contexto de redes SDN.

### 6.2.2 Seleção de contexto

O experimento foi realizado *in vitro*, no qual foi utilizada uma infraestrutura virtual para criação de um ambiente SDN, constituído por clientes, servidores, switches e controlador SDN.

### 6.2.3 Variáveis

As **variáveis independentes** do experimento foram: número de hosts geradores de fluxos de pacotes, protocolo do fluxo gerado (TCP ou UDP), carga do fluxo gerado, e parâmetros do modelo de permuta (porcentagem de empréstimo e frequência de coleta de estatísticas dos fluxos).

As **variáveis dependentes** foram: número de pacotes enviados, número de pacotes descartados, jitter (para o protocolo UDP), e latência.

### 6.2.4 Formulação de Hipóteses

A seguinte questão de pesquisa foi formulada a partir dos objetivos do estudo: O modelo de permuta proposto melhora o aproveitamento da largura de banda? Para avaliar essa questão foram utilizadas as métricas: número de pacotes enviados, número de pacotes descartados, jitter

(para o protocolo UDP), e latência. Sendo assim, com os objetivos, as questões e as métricas definidas, foram levantadas as seguintes hipóteses:

Em relação aos pacotes enviados **H0**: O uso do modelo de permuta e o não uso de um modelo resultam em número médio de pacotes enviados **iguais**. **H1**: O uso do modelo de permuta e o não uso de um modelo resultam em número médio de pacotes enviados **diferentes**.

Em relação aos pacotes descartados **H0**: O uso do modelo de permuta e o não uso de um modelo resultam em número médio de pacotes descartados **iguais**. **H1**: O uso do modelo de permuta e o não uso de um modelo resultam em número médio de pacotes descartados **diferentes**.

Em relação ao jitter **H0**: O uso do modelo de permuta e o não uso de um modelo resultam em jitters médios **iguais**. **H1**: O uso do modelo de permuta e o não uso de um modelo resultam em jitters médios **diferentes**.

Em relação ao latência **H0**: O uso do modelo de permuta e o não uso de um modelo resultam em latências médias **iguais**. **H1**: O uso do modelo de permuta e o não uso de um modelo resultam em latências médias **diferentes**.

### 6.2.5 Seleção de Objetos

Para diferenciar os tipos de tráfego na rede SDN foram considerados dois protocolos: TCP e UDP.

## 6.3 Design Experimental

Os seguintes passos são descritos para a execução do experimento:

1. Definir a topologia do experimento;
2. Para a utilização do modelo de permuta:
  - a) Definir parâmetro de percentual de compartilhamento;
  - b) Definir parâmetro de frequência de coleta de estatísticas dos fluxos nas filas;
  - c) Definir políticas de QoS das filas do link a ser compartilhado.
3. Para cada objeto experimental (sem modelo e com modelo de permuta):
  - a) Iniciar execução do objeto;
  - b) Por meio dos *hosts* clientes, injetar carga de pacotes TCP e UDP no link compartilhado, em direção ao host servidor;
  - c) Coletar as métricas experimentais.
4. Aplicar os testes estatísticos adequados para análise dos dados coletados.

## 6.4 Instrumentação

Os seguintes softwares foram utilizados para a criação do experimento:

- *Mininet* versão 2.2.1, um emulador de rede, para criação do ambiente SDN;
- *iPerf* versão 3, uma ferramenta de benchmark de rede, para a sintetização e injeção de pacotes TCP e UDP;
- *Floodlight* versão 1.0, um controlador SDN, para orquestrar a largura de banda das filas dos links do *switch* SDN;
- *Open vSwitch* versão 2.14.90, um *switch* SDN virtual, para comutação de pacotes e comunicação com o controlador SDN.

O seguinte hardware foi utilizado para a execução do experimento:

- Sistema Operacional Ubuntu Server 18.04 LTS;
- Disco rígido de 50GB de armazenamento;
- Processador com dois núcleos;
- 8 GB de memória RAM;

## 6.5 Operação do Experimento

Para a execução do experimento foi necessário a preparação do ambiente SDN, para tal, foi instalado no servidor supracitado o emulador de redes *Mininet*. Neste mesmo *hardware* foi instalado o controlador *Floodlight* modificado – com a adição do mecanismo de permuta.

### 6.5.1 Execução

O Quadro 1 exibe os ativos da topologia criada para o experimento. Cada linha do quadro representa um componente da topologia com alguns atributos para auxiliar no entendimento.

No momento da execução do experimento, seguindo o design experimental, foram realizados os seguintes passos e configurações:

1. A topologia da rede SDN, apresentada na Figura 24, contém 3 *hosts* clientes *iPerf* (H1, H3 e H4), um *host* servidor *iPerf* (H2), um comutador *Open vSwitch* (S1) conectado ao controlador *Floodlight*;
2. Os parâmetros para a utilização do modelo de permuta foram:

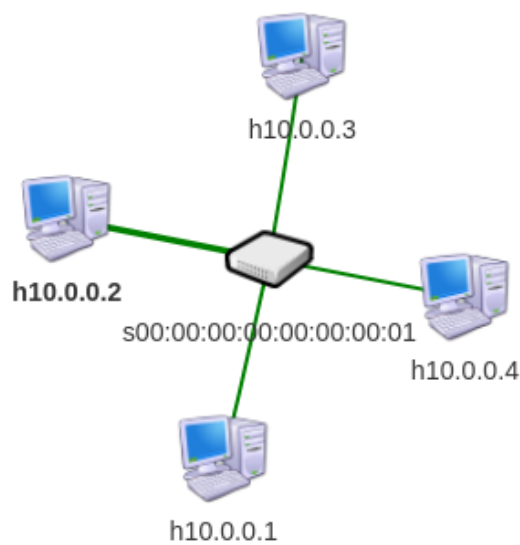


Quadro 1 – Elementos Topologia

ID	IP	Tipo
H1	10.0.0.1	Cliente
H2	10.0.0.2	Servidor
H3	10.0.0.3	Cliente
H4	10.0.0.4	Cliente
S1	-	switch

Fonte: (Autor, 2020)

Figura 24 – Topologia experimentos



Fonte: (Autor, 2020)

- a) Percentual de compartilhamento: 3%;
  - b) Frequência de coleta de estatísticas dos fluxos nas filas: uma coleta a cada 2 segundos;
  - c) Políticas de QoS das filas do link a ser orquestrado: foram definidas 3 filas, Q0 com 15Mbps, Q1 com 10Mbps e Q2 com 6Mbps em um link com tamanho total de 50Mbps;
3. Cada objeto experimental foi executado uma vez durante 60 segundos;
  4. Para cada *host* cliente apresentado na Figura 24, a carga de pacotes injetada na rede foi a seguinte:
    - a) H1: 100MB; 8Mbps; e protocolo UDP;
    - b) H3: 100MB; 8Mbps; e protocolo UDP;
    - c) H4: 100MB; 8Mbps; e protocolo UDP;

5. Os pacotes enviados a partir dos *hosts* clientes (H1, H3 e H4) para o *host* servidor (H2) foram encaminhados da seguinte forma:
  - a) De H1 para Q0;
  - b) De H3 para Q1;
  - c) De H4 para Q2;
6. Para cada objeto experimental, as métricas experimentais foram coletadas 31 vezes para que fosse possível realizar o cálculo de desvio padrão amostral.

### 6.5.2 Validação dos Dados

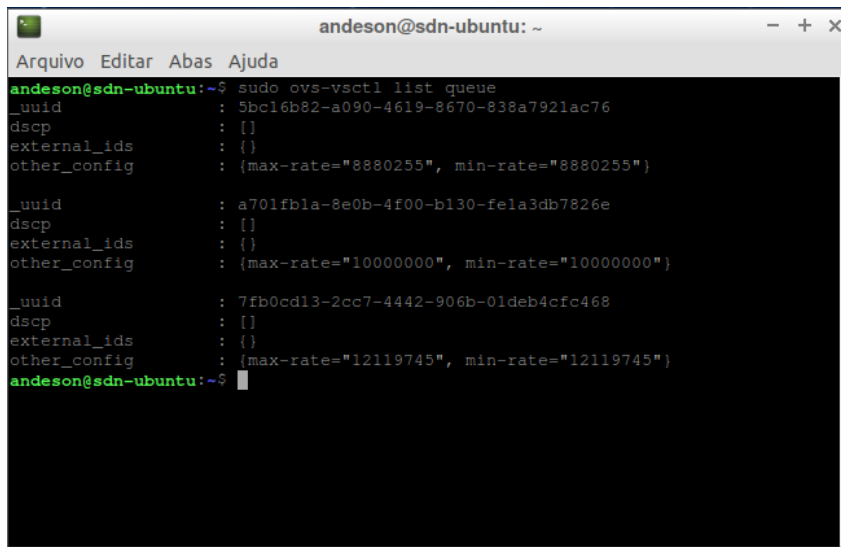
Durante a execução dos testes, com e sem o uso do mecanismo de compartilhamento de banda, houve perda de pacotes na fila Q2. Utilizando o mecanismo de compartilhamento, os tamanhos das filas Q2 e Q0 são alterados durante a execução, já que, pelo fato de Q0 possuir o maior tamanho de largura de banda disponível, Q2 toma por empréstimo parcelas da banda ociosa disponível de Q0 até que não haja mais perda de pacotes. As figuras Figura 25 e Figura 26, exibem, respectivamente, os tamanhos das filas especificados pelos atributos *min-rate* e *max-rate* no início da atuação do algoritmo, ou seja, antes de haver perda de pacotes e depois da execução de uma das baterias de testes, isto é, após a atuação do algoritmo realizando o processo de compartilhamento de largura de banda entre as filas.

Figura 25 – Tamanho da filas antes da perda de pacotes

```
anderson@sdn-ubuntu: ~  
Arquivo Editar Abas Ajuda  
anderson@sdn-ubuntu:~$ sudo ovs-vsctl list queue  
_uuid      : 5bc16b82-a090-4619-8670-838a7921ac76  
dscp       : []  
external_ids : {}  
other_config : {max-rate="6000000", min-rate="6000000"}  
  
_uuid      : a701fbl1a-8e0b-4f00-b130-fela3db7826e  
dscp       : []  
external_ids : {}  
other_config : {max-rate="10000000", min-rate="10000000"}  
  
_uuid      : 7fb0cd13-2cc7-4442-906b-01deb4cfc468  
dscp       : []  
external_ids : {}  
other_config : {max-rate="15000000", min-rate="15000000"}  
anderson@sdn-ubuntu:~$
```

Fonte: (Autor, 2020)

Figura 26 – Tamanho da filas após atuação do algoritmo de compartilhamento de banda



```
andeson@sdn-ubuntu: ~  
Arquivo Editar Abas Ajuda  
andeson@sdn-ubuntu:~$ sudo ovs-vsctl list queue  
_uuid      : 5bc16b82-a090-4619-8670-838a7921ac76  
dscp       : []  
external_ids : {}  
other_config : {max-rate="8880255", min-rate="8880255"}  
  
_uuid      : a701fbla-8e0b-4f00-b130-fe1a3db7826e  
dscp       : []  
external_ids : {}  
other_config : {max-rate="10000000", min-rate="10000000"}  
  
_uuid      : 7fb0cd13-2cc7-4442-906b-01deb4cfc468  
dscp       : []  
external_ids : {}  
other_config : {max-rate="12119745", min-rate="12119745"}  
andeson@sdn-ubuntu:~$
```

Fonte: (Autor, 2020)

Foram analisadas em relação à fila 2 (Q2), com e sem o mecanismo de permuta de largura de banda, as métricas:

- Quantidade total de pacotes enviados com sucesso;
- Quantidade total de pacotes enviados com erro;
- Vazão média.

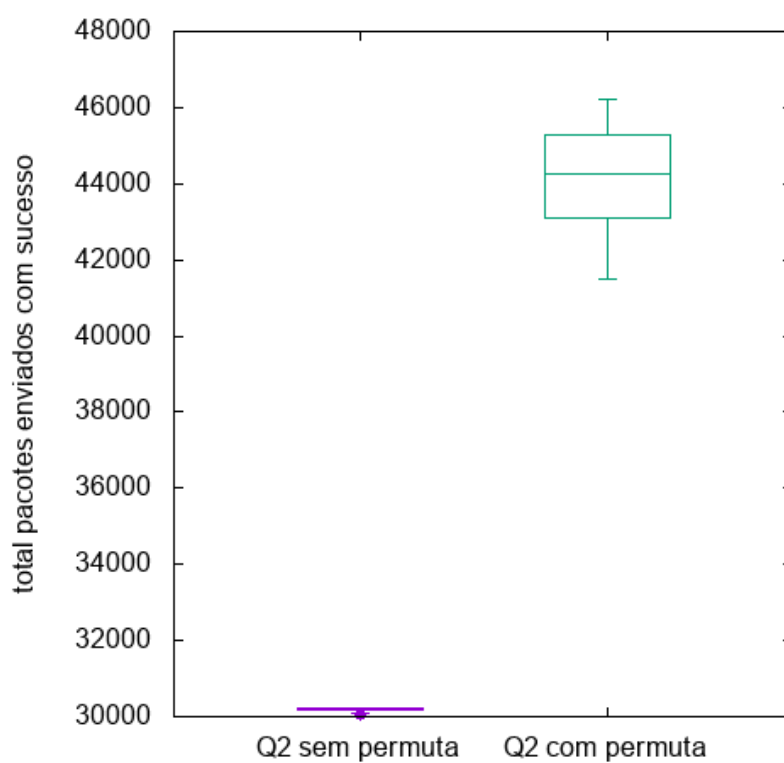
Para validação dos experimentos foram utilizados os seguintes testes estatísticos: *Kolmogorov-Smirnov* (KS), para verificar a normalidade dos dados; o teste de *Wilcoxon* para avaliar as hipóteses.

A Figura 27 apresenta a quantidade total de pacotes enviados com sucesso. Esse gráfico mostra que há uma diferença visualmente perceptível entre os dois tratamentos analisados, com o mecanismo de permutação (Q2 com permuta) e sem o mecanismo de permuta (Q2 sem permuta).

Para validar a diferença apresentada na Figura 27 foram aplicados testes estatísticos. Primeiro, foi analisada a normalidade dos dados coletados utilizando o teste KS. Para os dois tratamentos o teste KS retornou *p-value* igual a 0, ou seja, rejeitando fortemente a hipótese nula, o que indica que os dados para a métrica em questão não seguem uma distribuição normal. A Figura 28 mostra que os dados analisados não seguem uma distribuição normal.

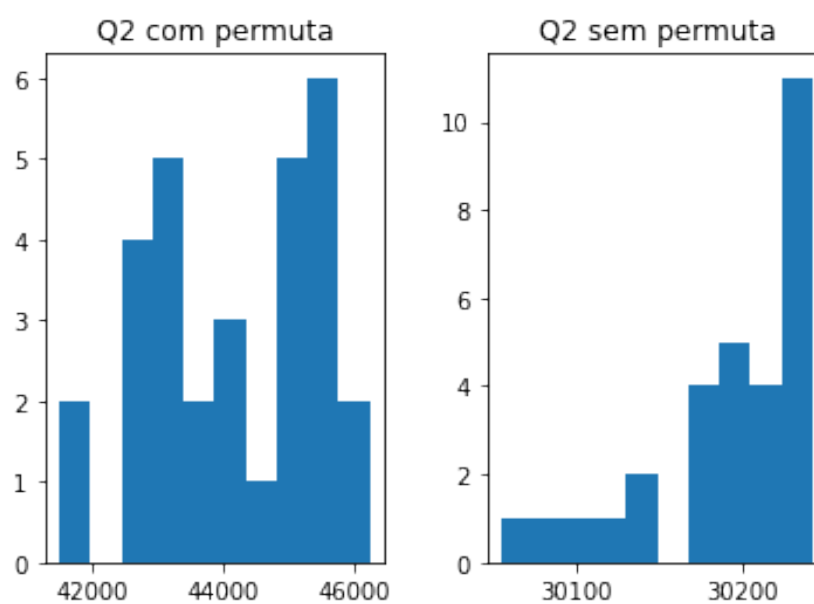
Como o teste KS mostrou que os dados não são normais, foi aplicado, em seguida, o teste não-paramétrico de *Wilcoxon* pareado. Sendo assim, a hipótese em relação à quantidade total de pacotes enviados com sucesso pode ser reformulada da seguinte forma:

Figura 27 – Total de pacotes enviados com sucesso.



Fonte: (Autor, 2020)

Figura 28 – Histograma pacotes enviados com sucesso

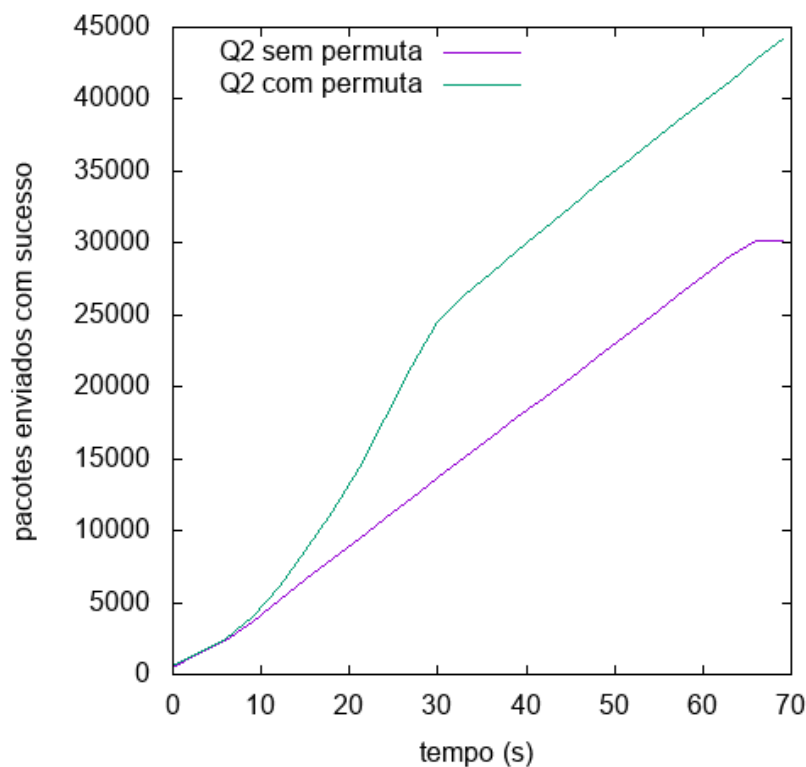


Fonte: (Autor, 2020)

- **H0**: O uso do modelo de permuta e o não uso de um modelo resultam em medianas de quantidade total de pacotes enviados com sucesso iguais.
- **H1**: O uso do modelo de permuta e o não uso de um modelo resultam em medianas de quantidade total de pacotes enviados com sucesso diferentes.

A aplicação do teste de *Wilcoxon* pareado resultou  $p\text{-value} \cong 1,734 \times 10^{-6} \ll 0,025$  (nível de significância de 0,05), rejeitando fortemente a hipótese nula. Portanto, como observado na Figura 27 e reforçado pelo teste de *Wilcoxon*, conclui-se que o uso do mecanismo de permuta aumenta significativamente a quantidade total de pacotes enviados com sucesso. Por fim, a Figura 29 também reforça a melhora com o mecanismo de permuta, apresentando a média do acúmulo de pacotes enviados com sucesso no tempo.

Figura 29 – Pacotes acumulados enviados com sucesso

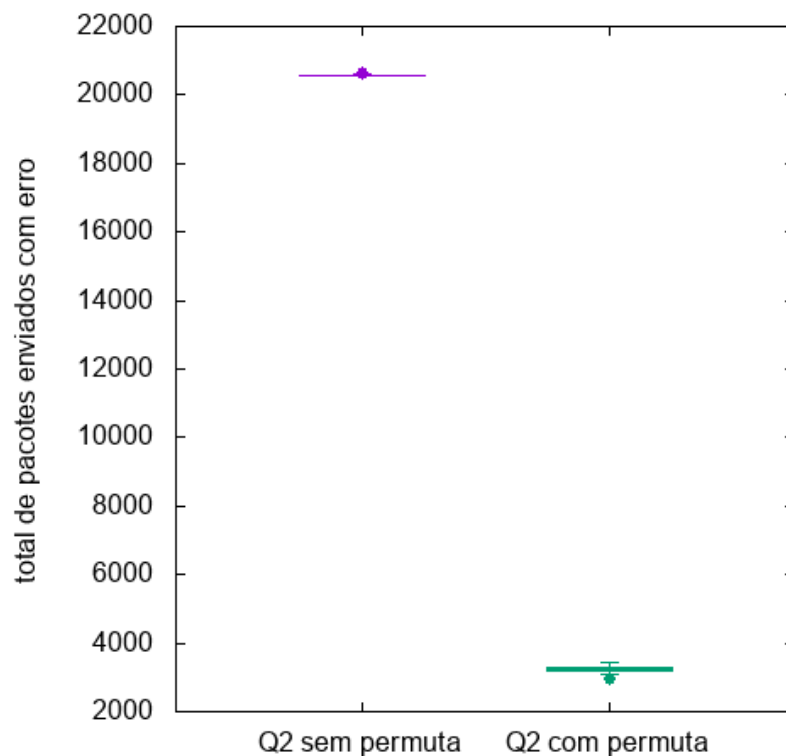


Fonte: (Autor, 2020)

A Figura 30 apresenta a quantidade total de pacotes enviados com erro, sem e com o mecanismo de permuta em funcionamento. Esse gráfico também mostra que há uma diferença visualmente perceptível entre os dois tratamentos analisados (Q2 com permuta e Q2 sem permuta).

Para validar a diferença apresentada na Figura 30 foram aplicados testes estatísticos. Primeiro, foi analisada a normalidade dos dados coletados utilizando o teste KS. Para os dois

Figura 30 – Distribuição da quantidade total de pacotes enviados com erro



Fonte: (Autor, 2020)

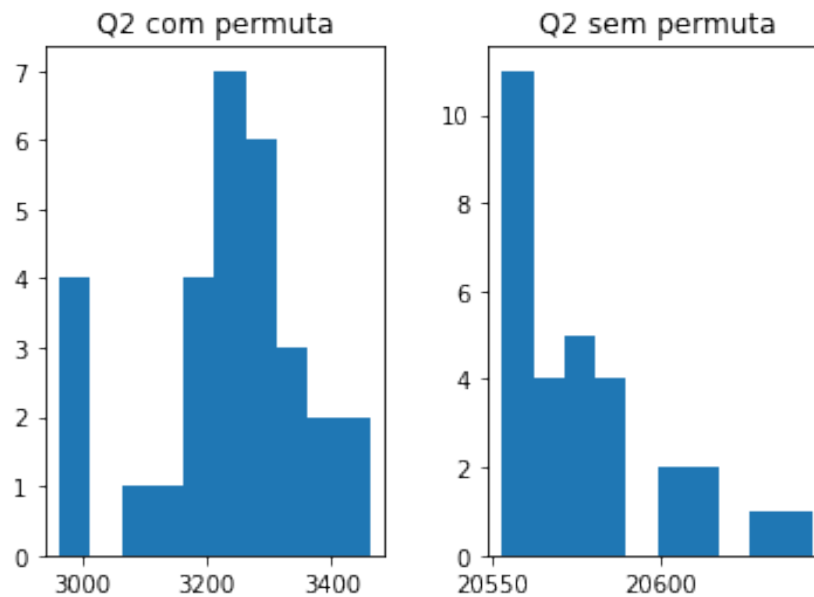
tratamentos o teste KS retornou *p-value* igual a 0, rejeitando a hipótese nula, o que indica que os dados para a métrica em questão também não seguem uma distribuição normal. A Figura 31 mostra que os dados analisados não seguem uma distribuição normal.

Como o teste KS mostrou que os dados não são normais, foi aplicado, em seguida, o teste não-paramétrico de *Wilcoxon* pareado. Sendo assim, a hipótese em relação à quantidade total de pacotes enviados com erro pode ser reformulada da seguinte forma:

- **H0:** O uso do modelo de permuta e o não uso de um modelo resultam em medianas de quantidade total de pacotes enviados com erro iguais.
- **H1:** O uso do modelo de permuta e o não uso de um modelo resultam em medianas de quantidade total de pacotes enviados com erro diferentes.

A aplicação do teste de *Wilcoxon* pareado resultou  $p\text{-valor} \cong 1,733 \times 10^{-6} \ll 0,025$  (nível de significância de 0,05), rejeitando fortemente a hipótese nula. Portanto, como observado na Figura 30 e reforçado pelo teste de *Wilcoxon*, conclui-se que o uso do mecanismo de permuta diminui significativamente a quantidade total de pacotes enviados com erro. Por fim, a Figura 32

Figura 31 – Histograma pacotes enviados com erro



Fonte: (Autor, 2020)

também reforça a melhora com o mecanismo de permuta, apresentando a média do acúmulo de pacotes enviados com erro no tempo.

A Figura 33 apresenta a distribuição dos dados de vazão média (Mbps). Esse gráfico mostra que há uma diferença consierável entre os dois tratamentos analisados (Q2 com permuta e Q2 sem permuta).

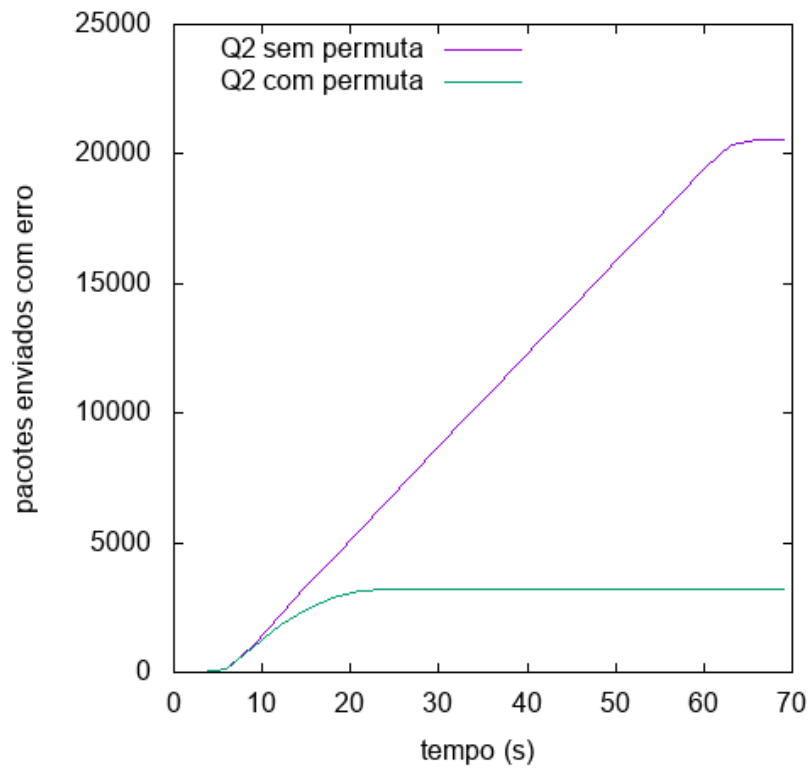
Para validar a diferença apresentada na Figura33 foram aplicados testes estatísticos. Primeiro, foi analisada a normalidade dos dados coletados utilizando o teste KS. Para os dois tratamentos o teste KS retornou  $p$ -value igual a 0, rejeitando fortemente a hipótese nula, indicando que os dados para a métrica em questão não seguem uma distribuição normal. A Figura 34 mostra que os dados dos tratamentos não seguem uma distribuição normal.

Como o teste KS mostrou que os dados não são normais, foi aplicado, em seguida, o teste não-paramétrico de *Wilcoxon* pareado. Deste modo, a hipótese em relação à vazão média pode ser reformulada da seguinte forma:

- **H0:** O uso do modelo de permuta e o não uso de um modelo resultam em medianas de vazão média iguais.
- **H1:** O uso do modelo de permuta e o não uso de um modelo resultam em medianas de vazão média diferentes.

A aplicação do teste de *Wilcoxon* pareado resultou  $p$ -value  $\cong 1,733 \times 10^{-6} \ll 0,025$  (nível de significância de 0,05), rejeitando a hipótese nula. Portanto, como observado na Figura

Figura 32 – Média do acúmulo de pacotes enviados com erro no tempo

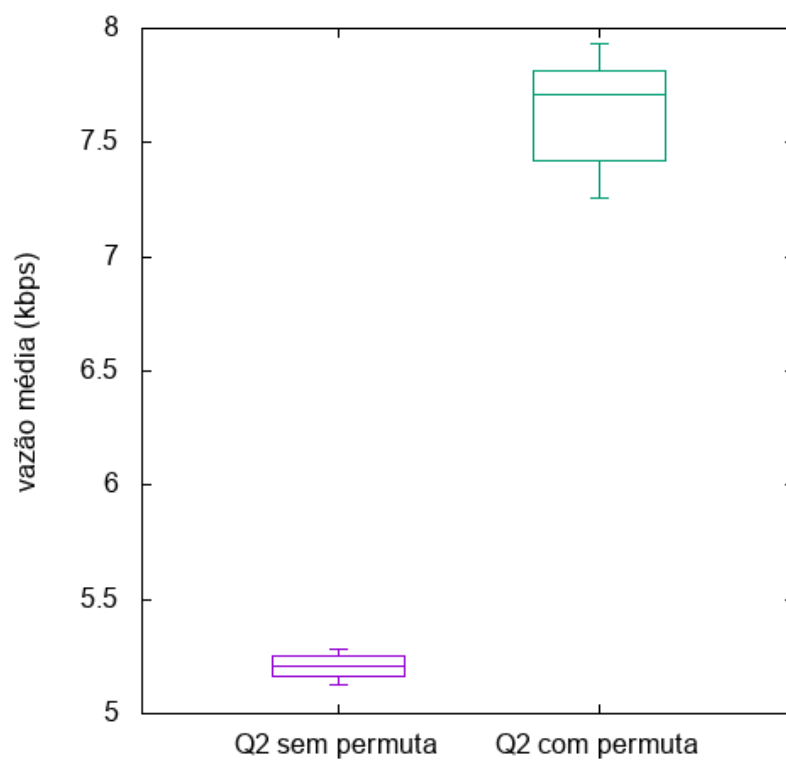


Fonte: (Autor, 2020)

33 e reforçado pelo teste de Wilcoxon, conclui-se que o uso do mecanismo de permuta aumenta significativamente a vazão média.

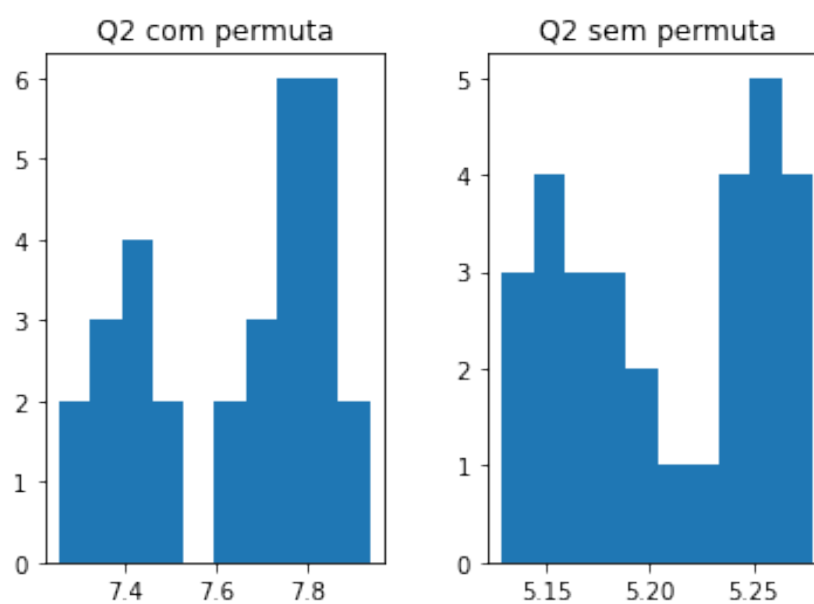


Figura 33 – Distribuição dos dados da vazão média



Fonte: (Autor, 2020)

Figura 34 – Histograma dos dados da vazão média



Fonte: (Autor, 2020)

# 7

## Conclusão

Neste trabalho foi discutido o compartilhamento de recursos em redes definidas por *software* utilizando modelos de alocação com critérios de justiça. Partindo desse contexto, foi desenvolvido um novo modelo de alocação, o MAMDFS (*MAM with Dynamic Fairness Sharing*), capaz de executar e gerenciar o compartilhamento de banda entre CTs distintas apoiando-se em conceitos de justiça para alcançar uma distribuição mais justa dos recursos disponíveis, aplicando políticas de QoS e minimizando a perda de informações durante o processo de transmissão de dados em um ambiente SDN.

Como parte deste trabalho também foi desenvolvido um módulo para o controlador de rede *Floodlight* implementando algoritmos baseados nos modelos de alocação de banda, para aplicar os conceitos e regras por estes definidas. Essas implementações, foram testadas em um ambiente de rede simulado e validadas através da aplicação de testes estatísticos, onde estes comprovaram que o uso do mecanismo de compartilhamento proposto pelo MAMDFS atingiu um considerável ganho de desempenho em relação ao MAM, elevando a eficiência do processo de transmissão de tráfego, ao ser capaz de gerenciar a largura de banda disponível entre as filas em momentos de congestionamento, minimizando assim a perda de pacotes.

Desse modo, concluiu-se que o serviço de compartilhamento de banda baseado em modelos de alocação com critérios de justiça implementados em controladores de redes SDN, atingiu um nível de ganho significativo e tem potencial para se tornar um serviço de grande valia nos processos de gerenciamento e controle de fluxos em ambientes de redes definidas por *software*. Diante disso, pode-se concluir também que a hipótese foi validada através dos experimentos e dos testes estatísticos aplicados aos resultados.

## 7.1 Contribuições

As contribuições deste trabalho foram principalmente: a elaboração de um modelo de alocação de recursos baseado em critérios de justiça, que tem como objetivo prover mecanismos de compartilhamento justo de largura de banda entre distintas classes de tráfego.

Outra contribuição deste trabalho foi, a adição de um módulo a um controlador de rede implementando o modelo proposto para validar a aplicabilidade de seus conceitos e que posteriormente pode ser utilizado em ambientes de redes reais.

Tanto a comunidade acadêmica quanto profissionais administradores de redes podem se beneficiar deste trabalho, uma vez que, os dados levantados na pesquisa podem auxiliar no desenvolvimento de novos projetos e profissionais podem fazer uso da solução implementada para criar e gerenciar políticas de QoS que possibilitem um melhor aproveitamento de recursos e melhorias de desempenho em infraestrutura de redes de computadores.

## 7.2 Dificuldades e limitações

Inicialmente, o intuito deste trabalho era de implementar um dos modelos de alocação de banda baseado no Valor de Shapley propostos por (Salgueiro, 2009). Porém, esse tema envolve cálculos matemáticos com elevado nível de complexidade, o que acabou gerando dificuldades durante o processo de desenvolvimento. Diante dessa situação, optou-se por implementar outro modelo e adotar a abordagem inicialmente pretendida em trabalhos futuros.

Para o desenvolvimento desta pesquisa, foi utilizado um cenário simplificado, com uma topologia contendo uma pequena quantidade de *hosts* e *switchs* em um ambiente simulado. Um cenário dessa natureza é muito menos complexo que um cenário em um ambiente real, onde podem haver inúmeros dispositivos de rede conectados a vários controladores. Sendo assim, pode-se considerar este fato uma limitação do trabalho.

Durante a fase de planejamento dos experimentos, foi elaborado um plano de execução em um cenário envolvendo equipamentos reais no laboratório do ELAN. Porém, por conta da suspensão dos trabalhos presenciais na Universidade Federal de Sergipe, devido à pandemia do novo coronavírus, todos os testes foram executados em um ambiente virtual. Portanto, isso pode ter limitado os resultados, uma vez que, a solução desenvolvida para este trabalho pode apresentar comportamento e resultados diferentes em um ambiente real.

## 7.3 Trabalhos futuros

Com o intuito de dar continuidade a este trabalho, o principal objetivo é dar prosseguimento nas pesquisas a fim de aperfeiçoar o modelo desenvolvido.

Realização de novos experimentos utilizando outros protocolos de rede e em ambientes reais. Serão realizados também, mais experimentos com outros parâmetros de porcentagem de permuta de largura de banda entre as filas, bem como outros valores para o intervalo de tempo para coleta das estatísticas. Enfim, execução de testes em vários cenários distintos a fim de realizar uma análise de desempenho mais aprofundada.

A implementação em controladores SDN de outros modelos de alocação de largura de banda.

Implementar mais recursos nos mecanismos de compartilhamento de largura de banda, adicionando algoritmos heurísticos com técnicas de predição para atestar se há ganho de desempenho.

Aplicar melhorias na solução e executar experimentos em ambientes variados, virtuais e reais, com intuito de produzir conteúdo para publicações de artigos científicos.

# Referências

- Openflow switch specification. Open, Networking Foundation, October 2013. URL <https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow/openflow-spec-v1.4.0.pdf>. Citado na página 23.
- aaaa. Nenhuma citação no texto.
- D. Adami, C. Callegari, S. Giordano, and M. Pagano. A new ns2 simulation module for bandwidth constraints models in ds-te networks. In *ICC Workshops - 2008 IEEE International Conference on Communications Workshops*, pages 251–255, 2008. doi: <10.1109/ICCW.2008.53>. Citado na página 27.
- A. V. Akella and K. Xiong. Quality of service (qos)-guaranteed network resource allocation via software defined networking (sdn). In *2014 IEEE 12th International Conference on Dependable, Autonomic and Secure Computing*, pages 7–13, Aug 2014. doi: <10.1109/DASC.2014.11>. Citado 2 vezes nas páginas 39 e 41.
- W. Aljoby, X. Wang, T. Z. J. Fu, and R. T. B. Ma. On sdn-enabled online and dynamic bandwidth allocation for stream analytics. *IEEE Journal on Selected Areas in Communications*, 37(8): 1688–1702, Aug 2019. ISSN 1558-0008. doi: <10.1109/JSAC.2019.2927062>. Citado 4 vezes nas páginas 37, 41, 46 e 49.
- J. Ash. Max Allocation with Reservation Bandwidth Constraints Model for Diffserv-aware MPLS Traffic Engineering & Performance Comparisons. (4126), 2005. URL <https://tools.ietf.org/html/rfc4126>. Citado 2 vezes nas páginas 28 e 29.
- D. Awduche, J. Malcolm, J. Agogbua, M. O’Dell, and J. McManus. Requirements for Traffic Engineering Over MPLS. (2702):1–29, 1999. URL <http://www.ietf.org/rfc/rfc2702.txt>. Citado na página 18.
- Basili, Victor R. The goal question metric approach. *University Of Maryland*, 1994. Citado 2 vezes nas páginas 60 e 61.
- Begen. Qoe-aware bandwidth broker for http adaptive streaming flows in an sdn-enabled hfc network. *IEEE Transactions on Broadcasting*, 64(2):575–589, June 2018. ISSN 1557-9611. doi: <10.1109/TBC.2018.2816789>. Citado 4 vezes nas páginas 38, 41, 47 e 49.
- J. M. Boley. Adaptive qos for data transfers using software-defined networking. In *2016 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, pages 1–6, Nov 2016. doi: <10.1109/ANTS.2016.7947874>. Citado 4 vezes nas páginas 37, 41, 46 e 49.
- G. L. R. d. Carvalho. *Compartilhamento de banda em redes definidas por software*. Universidade Federal de Sergipe, 2017. Citado na página 57.
- Y. Chang, Y. Chen, T. Chen, J. Chen, S. Chiu, and W. Chang. Software-defined dynamic bandwidth management. In *2019 21st International Conference on Advanced Communication Technology (ICACT)*, pages 201–205, Feb 2019. doi: <10.23919/ICACT.2019.8702052>. Citado 4 vezes nas páginas 37, 41, 47 e 49.

- Cisco. Redes definidas por software, 2020. URL <[https://www.cisco.com/c/pt\\_br/solutions/software-defined-networking/overview.html](https://www.cisco.com/c/pt_br/solutions/software-defined-networking/overview.html)>. Acessado em: 30/05/2020. Citado na página 17.
- Cisco Systems. Qos frequently asked questions, 2009. URL <<https://www.cisco.com/c/en/us/support/docs/quality-of-service-qos/qos-policing/22833-qos-faq.html>>. Citado na página 26.
- T. Couto. Requisitos não-funcionais em modelos de processos de negócio: Uma revisão sistemática. 05 2013. Citado na página 34.
- S. Dutt, D. Habibi, and I. Ahmad. A low cost atheros system-on-chip and openwrt based testbed for 802.11 wlan research. In *TENCON 2012 IEEE Region 10 Conference*, pages 1–4, 2012. doi: <10.1109/TENCON.2012.6412177>. Citado na página 48.
- F. L. Faucheur. Russian Dolls Bandwidth Constraints Model for Diffserv-aware MPLS Traffic Engineering. (4127), 2005. URL <<http://www.ietf.org/rfc/rfc4127.txt>>. Citado na página 29.
- F. L. Faucheur and W. Lai. Maximum Allocation Bandwidth Constraints Model for Diffserv-aware MPLS Traffic Engineering. (4125), 2005. URL <<http://www.ietf.org/rfc/rfc4125.txt>>. Citado na página 28.
- N. Feamster, J. Rexford, E. Zegura, and A. Morais. The road to sdn: An intellectual history of programmable networks. *SIGCOMM Comput. Commun. Rev.*, 44(2):87–98, Apr. 2014. ISSN 0146-4833. doi: <10.1145/2602204.2602219>. URL <<https://doi.org/10.1145/2602204.2602219>>. Citado na página 18.
- T. Feng and K. Wang. Joint allocation and scheduling of network resource for multiple control applications in sdn. In *2014 IEEE Network Operations and Management Symposium (NOMS)*, pages 1–7, May 2014. doi: <10.1109/NOMS.2014.6838242>. Citado 2 vezes nas páginas 38 e 41.
- P. Floodlight. Floodlight, 2020. URL <<https://floodlight.atlassian.net/wiki/spaces/floodlightcontroller/pages/1343647/Floodlight+Projects>>. Citado na página 54.
- Q. Fu, L. Qing, A. Yingzhu, and F. Yamei. A priority based virtual network bandwidth guarantee method in software defined network. In *2015 6th IEEE International Conference on Software Engineering and Service Science (ICSESS)*, pages 153–156, Sep. 2015. doi: <10.1109/ICSESS.2015.7339026>. Citado 2 vezes nas páginas 38 e 41.
- A. C. Gil. *Métodos e técnicas de pesquisa social*. 6 edição. Editora Atlas, 2008. Citado na página 21.
- F. Hu, B. Chen, X. Zhu, and H. Shen. Sdn-based efficient bandwidth allocation for caching enabled cognitive radio networks. In *2016 IEEE Trustcom/BigDataSE/ISPA*, pages 1382–1389, Aug 2016. doi: <10.1109/TrustCom.2016.0218>. Citado 2 vezes nas páginas 38 e 41.
- M. Karakus and A. Duresi. Quality of service (qos) in software defined networking (sdn): A survey. *Journal of Network and Computer Applications*, 80:200 – 218, 2017. ISSN 1084-8045. doi: <<https://doi.org/10.1016/j.jnca.2016.12.019>>. URL <<http://www.sciencedirect.com/science/article/pii/S1084804516303186>>. Citado na página 32.

- S. Kim. Cognitive radio bandwidth sharing scheme based on the two-way matching game. *Wirel. Pers. Commun.*, 68(3):893–905, Feb. 2013. ISSN 0929-6212. doi: <10.1007/s11277-011-0488-z>. URL <<http://dx.doi.org/10.1007/s11277-011-0488-z>>. Citado na página 17.
- B. Kitchenham. Procedures for performing systematic reviews. Keele university. technical report tr/se-0401, Department of Computer Science, Keele University, UK, 2004. Citado na página 33.
- J. F. Kurose and K. W. Ross. *Redes de Computadores e a Internet: Uma Abordagem Top-down*. Pearson do Brasil, 5ª edition, 2010. ISBN 8588639181. Citado na página 18.
- W. Lai. Bandwidth constraints models for differentiated services (diffserv)-aware mpls traffic engineering: Performance evaluation, 2005. URL <<https://tools.ietf.org/html/rfc4128>>. Citado na página 28.
- A. Lara and B. Ramamurthy. Opensec: A framework for implementing security policies using openflow. In *2014 IEEE Global Communications Conference*, pages 781–786, 2014. doi: <10.1109/GLOCOM.2014.7036903>. Citado na página 25.
- Z. Li, X. Zhu, J. Gahm, R. Pan, H. Hu, A. C. Begen, and D. Oran. Probe and adapt: Rate adaptation for http video streaming at scale. *IEEE Journal on Selected Areas in Communications*, 32(4): 719–733, 2014. doi: <10.1109/JSAC.2014.140405>. Citado na página 47.
- T. Lima, M. Júnior, and M. A. Nunes. *Mining on Line General Opinions About Sustainability of Hotels: A Systematic Literature Mapping*, pages 558–574. 07 2018. ISBN 978-3-319-95161-4. doi: <10.1007/978-3-319-95162-1\_38>. Citado na página 60.
- M. d. A. Marconi and E. M. Lakatos. *Fundamentos de metodologia científica. 5 edição*. Editora Atlas, São Paulo, 2003. Citado na página 21.
- N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner. Openflow: Enabling innovation in campus networks. *Computer Communication Review*, 38:69–74, 04 2008. doi: <10.1145/1355734.1355746>. Citado na página 17.
- F. P. D. Nascimento and F. L. L. Souza. *Metodologia da pesquisa científica teoria e prática: como elaborar TCC*. Fortaleza, 2017. Citado na página 21.
- B. A. A. Nunes, M. Mendonca, X. N. Nguyen, K. Obraczka, and T. Turletti. A survey of software-defined networking: Past, present, and future of programmable networks. *IEEE Communications Surveys Tutorials*, 16:1617–1634, 2014. Citado na página 17.
- J. A. M. d. Oliveira. *Modelos de Alocação de Banda com Justiça em Controladores de Redes Definidas por Software*. Universidade Federal de Sergipe, 2020. Citado na página 49.
- OpenDaylight. Opendaylight controller, 2020. URL <<https://wiki.opendaylight.org/>>. Citado na página 46.
- K. Petersen, R. Feldt, S. Mujtaba, and M. Mattsson. Systematic mapping studies in software engineering. In *EASE*, 2008. Citado na página 33.
- RFC-7047. The open vswitch database management protocol, 2013. URL <<https://tools.ietf.org/html/rfc7047>>. Citado na página 26.

- RYU. Ryu sdn framework, 2020. URL <<https://github.com/faucetsdn/ryu>>. Citado na página 47.
- E. Salgueiro. Alocação de Recursos com Justiça: Uma Aplicação de Jogos Cooperativos em Redes de Computadores. 20(2):205, 2009. doi: <10.1590/S0103-56652008000200018>. Citado 11 vezes nas páginas 18, 29, 30, 31, 36, 37, 45, 49, 50, 51 e 74.
- SDX. What is software defined networking (sdn)? definition, 2018. URL <<https://www.sdxcentral.com/networking/sdn/definitions/what-the-definition-of-software-defined-networking-sdn/>>. Citado na página 24.
- Sdx-Central. Sdn controller, 2018. URL <<https://www.sdxcentral.com/sdn/definitions/sdn-controllers/>>. Citado na página 24.
- S. Sezer, S. Scott-Hayward, P. K. Chouhan, B. Fraser, D. Lake, J. Finnegan, N. Viljoen, M. Miller, and N. Rao. Are we ready for sdn? implementation challenges for software-defined networks. *IEEE Communications Magazine*, 51(7):36–43, July 2013. ISSN 0163-6804. doi: <10.1109/MCOM.2013.6553676>. Citado na página 25.
- I. Suliman, C. Pomalaza-Ráez, I. Oppermann, and J. Lehtomäki. Radio resource allocation in heterogeneous wireless networks using cooperative games. In *Proc. Nordic Radio Symposium*, 2004. URL <[http://www.ee.oulu.fi/~jannel/PDFs/suliman\\_NRS04.pdf](http://www.ee.oulu.fi/~jannel/PDFs/suliman_NRS04.pdf)>. Citado na página 18.
- A. S. Tanenbaum. *Redes De Computadores*. Elsevier, 2003. Citado na página 27.
- S. Tang and S. Liu. A heuristic algorithm for optimal discrete bandwidth allocation in sdn networks. In *2015 IEEE Global Communications Conference (GLOBECOM)*, pages 1–7, Dec 2015. doi: <10.1109/GLOCOM.2015.7417174>. Citado 4 vezes nas páginas 37, 41, 46 e 49.
- N. Thazin and K. M. Nwe. End-to-end dynamic bandwidth resource allocation based on qos demand in sdn. In *2019 25th Asia-Pacific Conference on Communications (APCC)*, pages 244–249, Nov 2019. doi: <10.1109/APCC47188.2019.9026511>. Citado 4 vezes nas páginas 38, 41, 47 e 49.
- P. H. Thinh, P. N. Nam, N. H. Thanh, and T. T. Huong. Sdn– based dynamic bandwidth allocation for multiple video-on-demand players over http. In *2019 International Conference on Information and Communication Technology Convergence (ICTC)*, pages 163–168, Oct 2019. doi: <10.1109/ICTC46691.2019.8939834>. Citado 2 vezes nas páginas 38 e 41.
- E. Torres, R. Reale, L. Sampaio, and J. Martins. A sdn/openflow framework for dynamic resource allocation based on bandwidth allocation model. *IEEE Latin America Transactions*, 18(05): 853–860, May 2020. ISSN 1548-0992. doi: <10.1109/TLA.2020.9082913>. Citado 2 vezes nas páginas 38 e 41.
- J. von NEUMANN and O. MORGENSTERN. *Theory of Games and Economic Behavior*. Princeton University Press, 1972. Citado na página 30.
- J. Wang and P. Shi. Research on bandwidth control technology based on sdn. In *2018 2nd IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*, pages 705–708, May 2018. doi: <10.1109/IMCEC.2018.8469269>. Citado 4 vezes nas páginas 38, 41, 47 e 49.
- R. K. Yin. *Estudo de Caso: Planejamento e Métodos*. Bookman, 2014. Citado na página 21.



- G. Zhang, L. Cong, L. Zhao, K. Yang, and H. Zhang. Competitive Resource Sharing Based on Game Theory in Cooperative Relay Networks. *ETRI Journal*, 31(1):89–91, feb 2009. ISSN 1225-6463. doi: <10.4218/etrij.09.0208.0330>. URL <<http://etrij.etri.re.kr/etrij/journal/article/article.do?volume=31&issue=1&page=89>>. Citado na página 18.
- M. F. Zhani, H. Elbiaze, and F. Kamoun. Analysis and prediction of real network traffic. *Journal of Networks*, 4, 11 2009. doi: <10.4304/jnw.4.9.855-865>. Citado na página 46.
- Y. Zhao, B. Yan, J. Wu, and J. Zhang. Software-defined dynamic bandwidth optimization (sd-dbo) algorithm for optical access and aggregation networks. *Photonic Network Communications*, 31(2):251–258, Apr 2016. ISSN 1572-8188. doi: <10.1007/s11107-015-0540-0>. URL <<https://doi.org/10.1007/s11107-015-0540-0>>. Citado 4 vezes nas páginas 37, 41, 46 e 49.
- X. Zhu, F. Chan, K. Kambhatla, and S. Ramakrishnan. Sdn based qoe optimization for http-based adaptive video streaming. In *2015 IEEE International Symposium on Multimedia (ISM)*, pages 120–123, Dec 2015. doi: <10.1109/ISM.2015.53>. Citado 2 vezes nas páginas 38 e 41.